IMM 385
September 1970

Courant Institute of
Mathematical Sciences

Frontal Motion in the Atmosphere

Eli L. Turkel

Prepared under Contract N00014-67-A-0467-0016
with the Office of Naval Research NR 062-160
Distribution of this document is unlimited.

New York University

NR 062-160

IMM 385
September 1970

New York University

Courant Institute of Matematical Sciences

FRONTAL MOTION IN THE ATMOSPHERE

Eli L. Turkel

This report represents results obtained at the Courant Institute of Mathematical Sciences, New York University, with the Office of Naval Research, Contract N00014-67-A-0467-0016.

Table of Contents

ABSTRACT: The motion of frontal disturbances in the atmosphere is studied based on several nonlinear models proposed by Stoker. In the first model, the air is considered to be an incompressible fluid moving over a plane tangent to the rotating earth. The fluid consists of two layers and the density in each layer is assumed to be constant. The hydrostatic pressure law is then used to reduce this to a two space dimensional model. The boundary between these layers is a contact discontinuity and so instabilities may occur at this frontal surface.

To simplify this model,we assume that the dynamics of the perturbations in the upper warm air layer can be neglected. In this case only the motion of the cold air need be studied. The frontal surface intersects the horizontal ground in a curve, called the front, which is a free boundary for the cold air. Following the procedure of Kasahara, Isaacson and Stoker, we make a numerical study of this model using generalizations of the Lax-Wendroff scheme. The movement of the front is based on following the motion of material particles at the front. This study indicates the development of the occlusion process from an initially sinusoidal frontal pattern. Thus, we show that the qualitative features of the occlusion process depend only on the Coriolis and gravitational forces while the thermodynamic processes can be ignored.

Various initial and boundary conditions are considered, and a comparison of their effect on the occlusion process is made. In all cases, the cold front propagates faster than the warm front, and a relatively strong mass convergence flow exists behind the cold front only. A cyclonic circulation pattern also appears near the cold front. These facts suggest the occurrence of severe storms associated with cold fronts, but not with warm fronts in the atmosphere. The methods developed here have application to general free boundary problems in fluid dynamics.

The numerical study of these equations is still quite difficult and so a one space dimensional model is introduced. Numerical comparison with the two dimensional model shows that this simplified theory gives many of the important characteristics of the frontal motion for reasonable lengths of time.

The one dimensional model is then considered for a semi-infinite domain with constant initial and boundary conditions. The solution of this first order hyperbolic system is expanded in a formal perturbation series. The lowest order terms satisfy a quasi-linear homogeneous hyperbolic system of equations. This system can be analyzed by introducing the Riemann invariants as defined by Lax. Necessary and sufficient conditions for the existence of continuous global solutions are given. When these continuous solutions exist, they can be found explicitly for both subsonic and supersonic flows. The higher order systems are all linear nonhomogeneous equations which can be solved explicitly for the terms in

the expansion. Comparison of the series, through second order terms, with the numerical solution of the original system shows close agreement away from the boundary. These techniques can be used for all nonhomogeneous quasi-linear systems where the solution to the homogeneous system is known.

Introduction

In meteorology it is known that the weather in the middle lattitudes is conditioned to a considerable degree by events associated with the propagation of wave-like disturbances on a discontinuity surface between warm and cold air masses in the atmosphere. The intersection of the discontinuity surface with the earth is known as a front. In this paper we will be interested in following the motion of these fronts.

The importance of nonlinear effects in the dynamical equations of the cold front were first pointed out by Freeman (1952) [6], Abdullah (1949) [1], and Tepper (1952) [18] who applied the method of characteristics to solve nonlinear equations in one space dimension. Later Stoker (1953) [16] developed a two-layer model for the motion of a frontal surface. Whitham (1953) [20] made a qualitative study of this model which strongly indicated that the evolution of frontal cyclones might well follow the pattern that leads to the occlusion effect observed in nature. Stoker together with Kasahara and Isaacson (1964) [8,9] made some numerical calculations with a one-layer model and found the beginnings of the occlusion effect. One of the objects of this paper is to continue and extend the investigations of Kasahara, Isaacson and Stoker (abbreviated as K.I.S.).

In Chapter II we will present several finite difference schemes for dealing with the problem under various initial and

boundary conditions. In the one layer theory we assume that the motion of the warm air is not affected by the motion of the cold air and hence the warm air remains in its initial state. This is based on the following reasoning of Stoker [17]. If the front develops a bulge the warm air can compensate for this by a slight change in its vertical component and with hardly any change in the horizontal components u and v. Since we are ignoring changes in the vertical velocity in any case, it is reasonable to assume that the warm air is unaffected by these movements of the front. However, in the cold air it is evident that relatively large changes in the components, u and v, of the velocity may be needed to move around such a frontal disturbance.

In the one layer model the discontinuity surface between the warm and cold air masses is a free surface. The motion of this free surface is determined solely by the dynamics of the cold air. However, in the two layer model this free surface becomes a contact discontinuity separating the warm and cold air masses. It is well known that the occurrence of contact discontinuities leads to instabilities both physically and numerically (e.g. see Richtmyer [14]). In Chapter III we will give some preliminary results using the two layer model.

In Chapter II we will discuss the one layer model. We are able to continue the solution for longer times that K.I.S. was able to. This allows us to achieve greater

insights into the formation of occluded fronts. It is planned to incorporate this theory into a large scale model for the circulation of the atmosphere. The motion of the fronts could then be followed by using a refined mesh in the neighborhood of the front. Ciment [4] has shown that such difference schemes with uneven mesh spacings are stable, in the linear case, for dissipative schemes as the Lax-Wendroff method.

Even this simplified model is very complicated and the equations can only be solved numerically and with considerable difficulty. Therefore, Stoker [16] introduced another model which contains only one space dimension but four dependent variables. This is done by introducing a long wave theory in the horizontal plane. This model is derived in Chapter I since the hypotheses of this model can be used to derive boundary conditions for the two dimensional model.

CHAPTER I. DERIVATION OF THE EQUATIONS

1. Two dimensional - Two layer theory

The earth is considered to be covered by two layers of air, the upper and lower layers corresponding respectively to the warm and cold air masses. The density of the air within each of these layers is assumed to be constant. Throughout this paper we consider a tangent plane approximation to the earth. The x-axis is the normal to this plane, the positive direction being directed away from the earth. The x and y axes are in the plane of the earth with the positive x axis directed to the East and the positive y axis to the North. Thus, we are ignoring the sphericity of the earth even though we shall not ignore its spin. Because of the earth's spin we shall assume that the coordinate system is rotating about the z axis with a constant angular velocity $\Omega = \omega \sin \phi = \frac{1}{2} f$, where ω is the angular velocity of the earth, ϕ is the latitude of the origin of the system and f is the (constant) Coriolis parameter.

Initially the cold air lies in a wedge under the warm air and the discontinuity surface between the two layers is inclined at angle α to the horizontal as shown in Figures 1a and 1b. The discontinuity surface is inclined with respect to the horizontal plane because of the Coriolis effect. However, the angle α is quite small, of the order of $\frac{1}{2}°$. The velocity in the warm air is that of the prevailing

westerlies while in the cold air the velocity is approximately that of the polar wind. Thus across the discontinuity surface there are tangential discontinuities in the velocities as well as jumps in the densities.

Following the presentation of Stoker [16] we begin with the Euler equations for fluid dynamics.

$$(1.1)\quad \begin{aligned} &(a)\quad \rho \frac{du}{dt} = -\frac{\partial p}{\partial x} + \rho F_{(x)} \\ &(b)\quad \rho \frac{dv}{dt} = -\frac{\partial p}{\partial y} + \rho F_{(y)} \qquad \frac{d}{dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z} \\ &(c)\quad \rho \frac{dw}{dt} = -\frac{\partial p}{\partial z} + \rho F_{(z)} \ . \end{aligned}$$

F is due to both gravitational and Coriolis forces. In addition we assume that the air is incompressible and of constant density in each layer and so we have

$$(d)\qquad \frac{\partial u}{\partial t} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \ .$$

We shall call this problem I.

In dynamic meteorology a standard assumption is the hydrostatic pressure law. This states that the vertical accelerations of the Coriolis term in the third equation of (1) can be ignored and hence

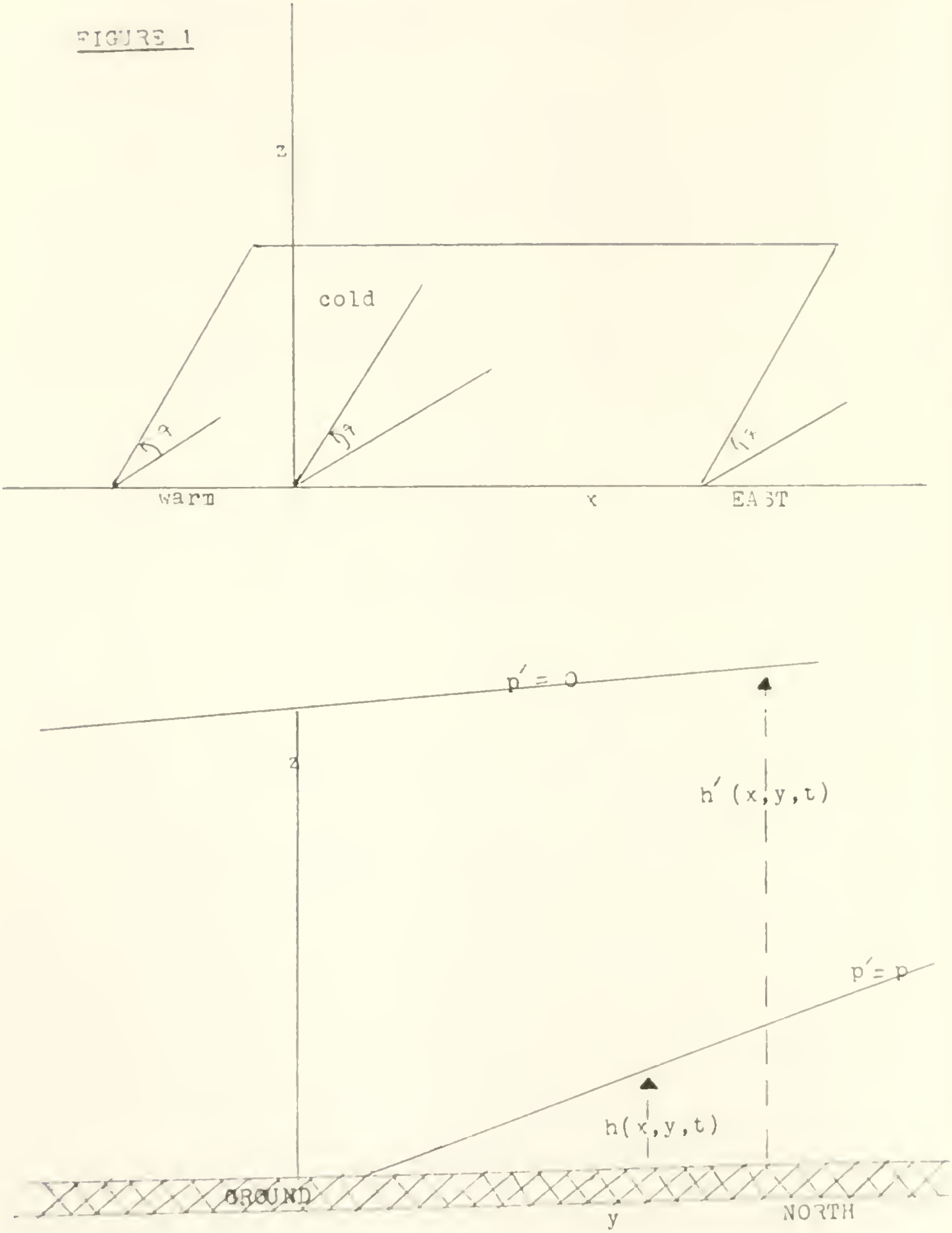
$$(1.2)\qquad \frac{\partial p}{\partial z} = -\rho g$$

Thus, on purely kinematic grounds we have

$$u = u(x,y,t) \ , \quad v = v(x,y,t), \quad w = 0.$$

Thus we have

FIGURE 1



$$(1.3)\quad \begin{aligned} &(a)\quad u_t + uu_x + vu_y = -\frac{1}{\rho} P_x + F_{(x)} \\ &(b)\quad v_t + uv_x + vv_y = -\frac{1}{\rho} P_y + F_{(y)} \\ &(c)\quad u_x + v_y = 0 , \end{aligned}$$

where

$$\begin{aligned} F_{(x)} &= 2\omega \sin\phi \cdot v = fv \\ F_{(y)} &= -2\omega \sin\phi \cdot u = -fu \end{aligned} \qquad f \text{ a constant}$$

Before continuing we wish to introduce notation to distinguish between the warm and cold air layers. Thus, from now on primed variables e.g. u', v' ref r to the warm air mass while unprimed variables e.g. u, v refer to the cold air.

We now integrate equation (1.2) $\frac{\partial p}{\partial z} = -\rho g$. Let $h = h(x,y,t)$ be the vertical height of the discontinuity surface between the warm and cold air and let $h' = h'(x,y,t)$ be the total height of the warm air. Assume $p' = 0$ at the top of the warm air. We then have

$$(1.4)\quad \begin{aligned} p'(x,y,z,t) &= \rho' g(h'-z) \\ p\ (x,y,z,t) &= \rho' g(h'-h) + \rho\, g(h-z) , \end{aligned}$$

where we have used the continuity condition $p' = p$ at $z = h$. Substituting (1.4) into (1.3) we get

$$(1.5)\quad \begin{aligned} &(a)\quad u_t + uu_x + vu_y = -g[\frac{\rho'}{\rho} h'_x + (1-\frac{\rho'}{\rho})h_x] + fv \\ &(b)\quad v_t + uv_x + vv_y = -g[\frac{\rho'}{\rho} h'_y + (1-\frac{\rho'}{\rho})h_y] - fu \\ &(c)\quad u_x + v_y = 0 \end{aligned}$$

(d) $u'_t + u'u'_x + v'u'_y = -gh'_x + fv'$

(e) $v'_t + u'v'_x + v'v'_y = -gh'_y - fu'$

(f) $u'_x + v'_y = 0$.

The kinematic condition at the free surfaces $z = h$ and $z' = h$ are $\frac{d}{dt}(z-h) = 0$, $\frac{d}{dt}(z-h') = 0$. Since $\frac{dz}{dt} = 0$ because we are ignoring vertical velocities, we can rewrite these equations using partial derivatives instead of particle derivatives and get

$$uh_x + vh_y + h_t = 0$$

$$u'h_x + v'h_y + h_t = 0$$

$$u'h'_x + v'h'_y + h'_t = 0 .$$

Combining these equations together with (1.5c,f) we get

$$(1.6) \qquad \begin{aligned} &h_t + (hu)_x + (hv)_y = 0 \\ &(h'-h)_t + [(h'-h)u]_x + [(h'-h)v]_y = 0 . \end{aligned}$$

Physically equations (1.6) are simply the equations for the conservation of mass in both the warm and cold air. If we substitute (1.6) into (1.5) in place of equations (1.5c,f) we arrive at problem II.

(a) $u_t + uu_x + vu_y = -g[\frac{\rho'}{\rho} h'_x + (1- \frac{\rho'}{\rho})h_x] + fv$

(b) $v_t + uv_x + vv_y = -g[\frac{\rho'}{\rho} h'_y + (1- \frac{\rho'}{\rho})h_y] - fu$

(c) $h_t + (hu)_x + (hv)_y = 0$

(1.7), II

(d) $\quad u'_t + u'u'_x + v'u'_y = -gh'_x + fv'$

(e) $\quad v'_t + u'v'_x + v'v'_y = -gh'_y - fu'$

(f) $\quad (h'-h)_t + [(h'-h)u]_x + [(h'-h)v]_y = 0$

System II has an exact solution corresponding to an initial state of a stationary front

$$(1.8)\quad \begin{aligned} &(a) \quad u' = \bar{u}' \\ &\qquad\; v' = 0 \\ &\qquad\; h' = -\frac{fu'}{g}(y+K_1) \\ &(b) \quad u = \bar{u} \\ &\qquad\; v = 0 \\ &\qquad\; h = \frac{f}{g(1-\frac{\rho'}{\rho})}\left(\frac{\rho'}{\rho}\bar{u}'-\bar{u}\right)(y-K_2) \end{aligned}$$

b. <u>Two-dimensional - Single layer theory</u>

The two layer theory just given involves several numerical difficulties. First, there are six dependent variables. Since the problem involves two space dimensions we require large matrices to record all the known values of these variables at all the mesh points and for some time step. Because of this the mesh must be fairly coarse. Another difficulty is the high sound speed in the warm air. The largest sound speed of the two layer model is of the

order of $\sqrt{\frac{g\rho'h'}{\rho}}$, since $\frac{\rho'}{\rho} \simeq 1$ and $h' >> h$ this sound speed is considerably larger than that of the cold air which is of the order of $\sqrt{g(1-\frac{\rho'}{\rho})h}$. By the Courant-Freidrichs-Lewy theory [5] the maximum time step allowable for stability is inversely proportional to the maximum sound speed. Thus the time steps must be very small and so the solution would require large amounts of computer time. However, since most of the dynamics is in the cold air the sound speed of physical relevance is that of the cold air. Thus the small time step is necessary for numerical rather than physical reasons. A final and more serious difficulty is that the discontinuity surface between the warm and cold air masses is a contact discontinuity. Thus Rayleigh and Helmholtz instabilities may occur. Even in the absence of physical instabilities various numerical instabilities occur in the neighborhood of contact discontinuities.

For these various reasons it becomes desirable to eliminate the influence of the warm air on the cold air. The physical justification of this assumption was discussed in the Introduction. Problem III is gotten by assuming that the perturbations in the cold air do not affect the warm air. Thus the initial conditions (1.8a) hold for all time. Substituting (1.8a) into system II we have system III.

$$
\begin{aligned}
& u_t + uu_x + vu_y + g(1-\frac{\rho'}{\rho})h_x = fv \\
(1.9),\ \text{III} \qquad & v_t + uv_x + vv_y + g(1-\frac{\rho'}{\rho})h_y = f(u-\frac{\rho'}{\rho}\bar{u}') \\
& h_t + (hu)_x + (hv)_y = 0 \ .
\end{aligned}
$$

In this system there are three dependent variables, the sound speeds are $c^2 = g(1- \frac{\rho'}{\rho})h$ and the front is a free surface. It was this problem that Kasahara, Isaacson and Stoker discussed in their paper.

c. <u>One dimensional model</u>

Stoker [16] developes problem IV as an approximation based on assuming waves of wave lengths that are large compared to the typical north-south lengths. Let $\eta(x,t)$ be the displacement of the front from the rigid wall* $y = \delta$ as shown in Figure 2. The velocity $v(x,y,t)$ is assumed to be zero at the rigid wall and then increase linearly to its value at the front $y = \delta - \eta(x,t)$ where it is denoted by $\bar{v}(x,t)$. h is zero at the front and increases linearly in y till the rigid wall $y = \delta$. Also, we denote the intersection of the discontinuity surface $z = h(x,y,t)$ with the plane $y = \delta$ as $\bar{h}(x,t)$. Finally we assume that u is independent of y and hence $u = u(x,t)$. Thus we have

(1.10)

(a) $$h(x,y,y) = \frac{y-\delta+\eta(x,t)}{\eta(x,t)} \bar{h}(x,t)$$

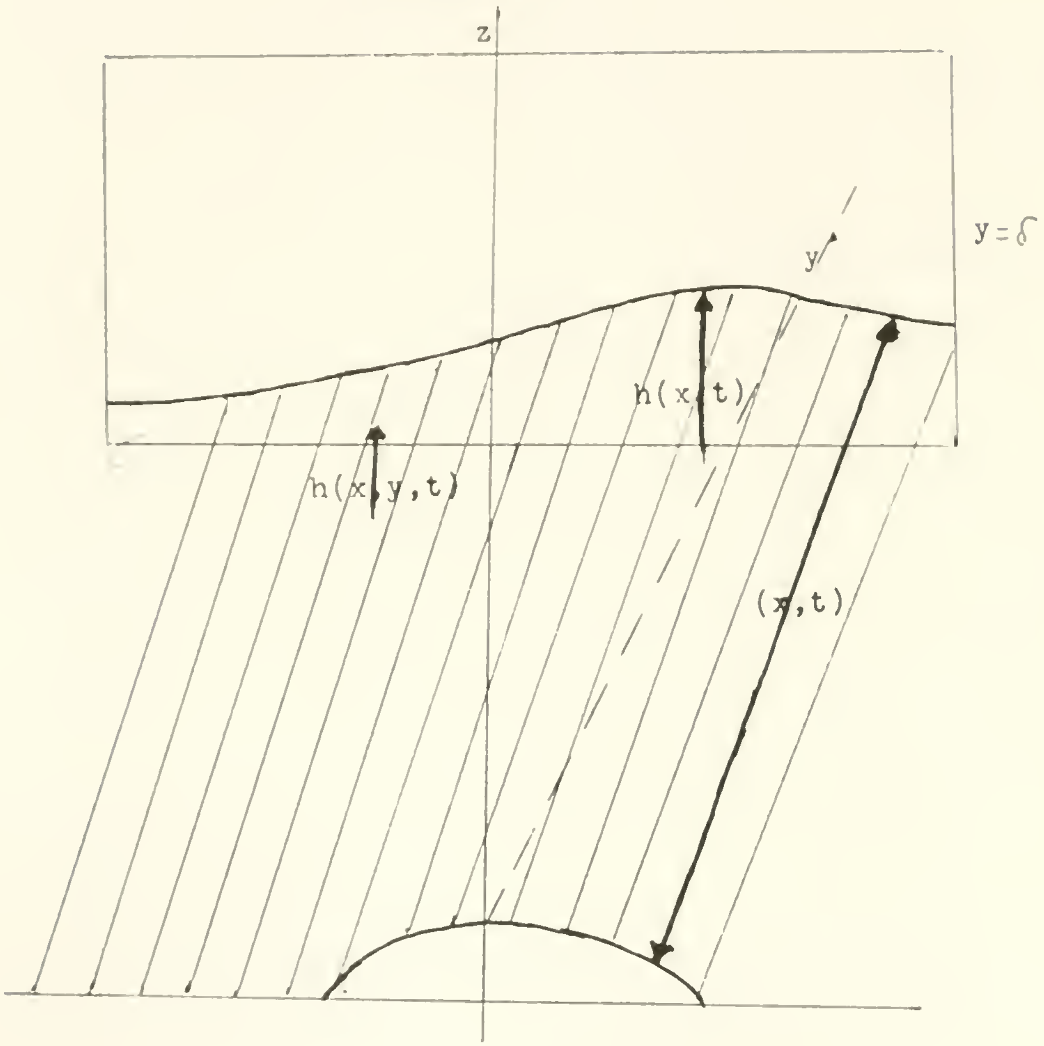
(b) $$v(x,y,t) = \frac{\delta - y}{\eta(x,t)} \bar{v}(x,t)$$

In addition we assume that a particle that starts on the front $y = \delta - \eta(x,t)$ remains on the front and so

(c) $$\bar{v}(x,t) = -(\eta_t + u\eta_x) .$$

* This is slightly different from Stoker's definition and corresponds to $\delta - \eta$ in his book.

FIGURE 2



$\eta(x,t)$ is the distance from the front to the wall $y = \delta$.

$\bar{h}(x,t)$ is the height, $h(x,y,t)$ at the wall $y = \delta$.

We now integrate the equations in system III from $y = \delta - \eta$ to $y = \delta$ and obtain system IV. To show how this is done we calculate several integrals explicitly.

$$\int_{\delta-\eta}^{\delta} h \, dy = \frac{\bar{h}}{\eta} \int_{\delta-\eta}^{\delta} [y-(\delta-\eta)] \, dy = \frac{1}{2} \bar{h} \eta$$

$$\int_{\delta-\eta}^{\delta} v \, dy = \frac{\bar{v}}{\eta} \int_{\delta-\eta}^{\delta} (\delta-y) \, dy = \frac{1}{2} \bar{v} \eta$$

We now differentiate these formulas keeping in mind that the limits of integration involve η which is a function of x and t. Thus we have

$$\frac{\partial}{\partial x} \int_{\delta-\eta}^{\delta} v \, dy = \int_{\delta-\eta}^{\delta} v_x \, dy - v(x,\delta-\eta,t) \frac{\partial}{\partial x}(\delta-\eta)$$

$$= \int_{\delta-\eta}^{\delta} v_x \, dy + \bar{v} \eta_x$$

So

$$\int_{\delta-\eta}^{\delta} v_x \, dy = \frac{\partial}{\partial x} \int_{\delta-\eta}^{\delta} v \, dy - \bar{v} \eta_x = \frac{1}{2} \frac{\partial}{\partial x} (\bar{v} \eta) - \bar{v} \eta_x$$

$$= \frac{1}{2} \bar{v}_x \eta + \frac{1}{2} \bar{v} \eta_x - \bar{v} \eta_x$$

or

$$\int_{\delta-\eta}^{\delta} v_x \, dy = \frac{1}{2} \bar{v}_x \eta - \frac{1}{2} \bar{v} \eta_x .$$

Similarly

$$\int_{\delta-\eta}^{\delta} h_x \, dy = \frac{1}{2} \bar{h}_x \eta + \frac{1}{2} \bar{h} \eta_x \qquad \text{since } h(x,\delta-\eta,t) = 0.$$

$$\int_{\delta-\eta}^{\delta} v_t \, dy = \frac{1}{2}\bar{v}_t \eta - \frac{1}{2}\bar{v}\eta_t$$

$$\int_{\delta-\eta}^{\delta} h_t \, dy = \frac{1}{2}\bar{h}_t \eta + \frac{1}{2}\bar{h}\eta_t$$

Also,

$$\int_{\delta-\eta}^{\delta} (hv)_y \, dy = hv\Big|_{\delta-\eta}^{\delta} = 0 \quad \text{since } h(x,\delta-\eta,t)=v(x,\delta,t)=0$$

$$\int_{\delta-\eta}^{\delta} (hu)_x \, dy = \frac{\partial}{\partial x}\Big[u \int_{\delta-\eta}^{\delta} h \, dy\Big]$$

$$= \frac{1}{2}(\bar{h}_x u + \bar{h}u_x)\eta + \frac{1}{2}\bar{h}u\eta_x$$

$$= \frac{1}{2}(\bar{h}u)_x \eta + \frac{1}{2}\bar{h}u\eta_x \, .$$

We now integrate system III with respect to y from $\delta-\eta$ to δ and then divide the resulting equations by η. Let $k = g(1 - \frac{\rho'}{\rho})$ and we have

$$u_t + uu_x + \frac{1}{2}k\bar{h}_x + \frac{1}{2}\frac{k\bar{h}}{\eta}\eta_x = \frac{1}{2}f\bar{v}$$

(1.11) $$\bar{v}_t + u\bar{v}_x - \frac{\bar{v}}{\eta}[\eta_t + u\eta_x] = \frac{\bar{v}^2}{\eta} - \frac{2k\bar{h}}{\eta} - 2f(u - \frac{\rho'}{\rho}\bar{u}')$$

$$\bar{h}_t + (\bar{h}u)_x + \frac{\bar{h}}{\eta}(\eta_t + u\eta_x) = 0 \, .$$

If we also include equation (1.10c) and use it to simplify the above equations we arrive at system IV.

$$
(1.12),\mathrm{IV}\qquad
\begin{aligned}
&(a)\quad u_t + uu_x + \frac{1}{2}k\bar{h}_x + \frac{1}{2}\frac{k\bar{h}}{\eta}\eta_x &&= \frac{1}{2}f\bar{v}\\
&(b)\quad \bar{h}_t + (\bar{h}u)_x &&= \frac{\bar{h}\bar{v}}{\eta}\\
&(c)\quad \bar{v}_t + u\bar{v}_x &&= -\frac{2k\bar{h}}{\eta} - 2f(u - \frac{\rho'}{\rho}\bar{u}')\\
&(d)\quad \eta_t + u\,\eta_x &&= -\bar{v}
\end{aligned}
$$

It is convenient to change variables by introducing the sound speed $c^2 = \frac{1}{2}kh$, we then have

$$
(1.13)\qquad
\begin{aligned}
&(a)\quad u_t + uu_x + 2cc_x + \frac{c^2}{\eta}\eta_x &&= \frac{1}{2}f\bar{v}\\
&(b)\quad c_t + \frac{c}{2}u_x + uc_x &&= \frac{1}{2}\frac{c\bar{v}}{\eta}\\
&(c)\quad \bar{v}_t + u\bar{v}_x &&= -4\frac{c^2}{\eta} - 2f(u - \frac{\rho'}{\rho}\bar{u}')\\
&(d)\quad \eta_t + u\,\eta_x &&= -\bar{v}
\end{aligned}
$$

The scheme is discussed in detail by Turkel [19]. Numerical solution of this model shows qualitative agreement with the single layer theory, Problem III, through the first eight hours. This system is also considered in a semi-infinite domain and a formal perturbation series solution is obtained. The lowest order term is analyzed by use of the Lax theory of Riemann invariants for systems of equations [12]. The first few terms of this series shows close agreement with the numerical solution of the system of equations.

CHAPTER II. TWO DIMENSIONAL - SINGLE LAYER THEORY

a. Initial conditions

In this chapter we discuss problem III, given by equations (1.9). This model is a two-space dimensional problem in which we assume that the warm air remains in its initial state for all time and hence we can concern ourselves with the motion of the cold air only. We assume that the cold air lies above a region D of the x-y plane and that this region is bounded on three sides by straight lines and on the fourth side by a curve C(t) as illustrated in Figure 3. The warm air lies above the cold air in three dimensional space and occupies the entire rectangle R in the x-y plane. The curve C is the line where the cold air ends i.e. where $h = 0$. According to the Glossary of Meteorology (American Meteorological Society 1959) the intersection of the discontinuity surface between the cold air and warm air with the earth's surface is called a surface front. However we shall follow popular usage and call C the front.

In this discussion we confine ourselves to a region D called the cold air mass domain. The curve $C(t):(x_C(t),y_C(t))$ is defined as the line along which $h = 0$. The points of C move in time with the velocity (u_C,v_C) of the particles on the front, and hence the following conditions hold along C:

$$h = 0$$

$$\frac{d}{dt}(x(t)) = u_C(x_C, y_C, t)$$

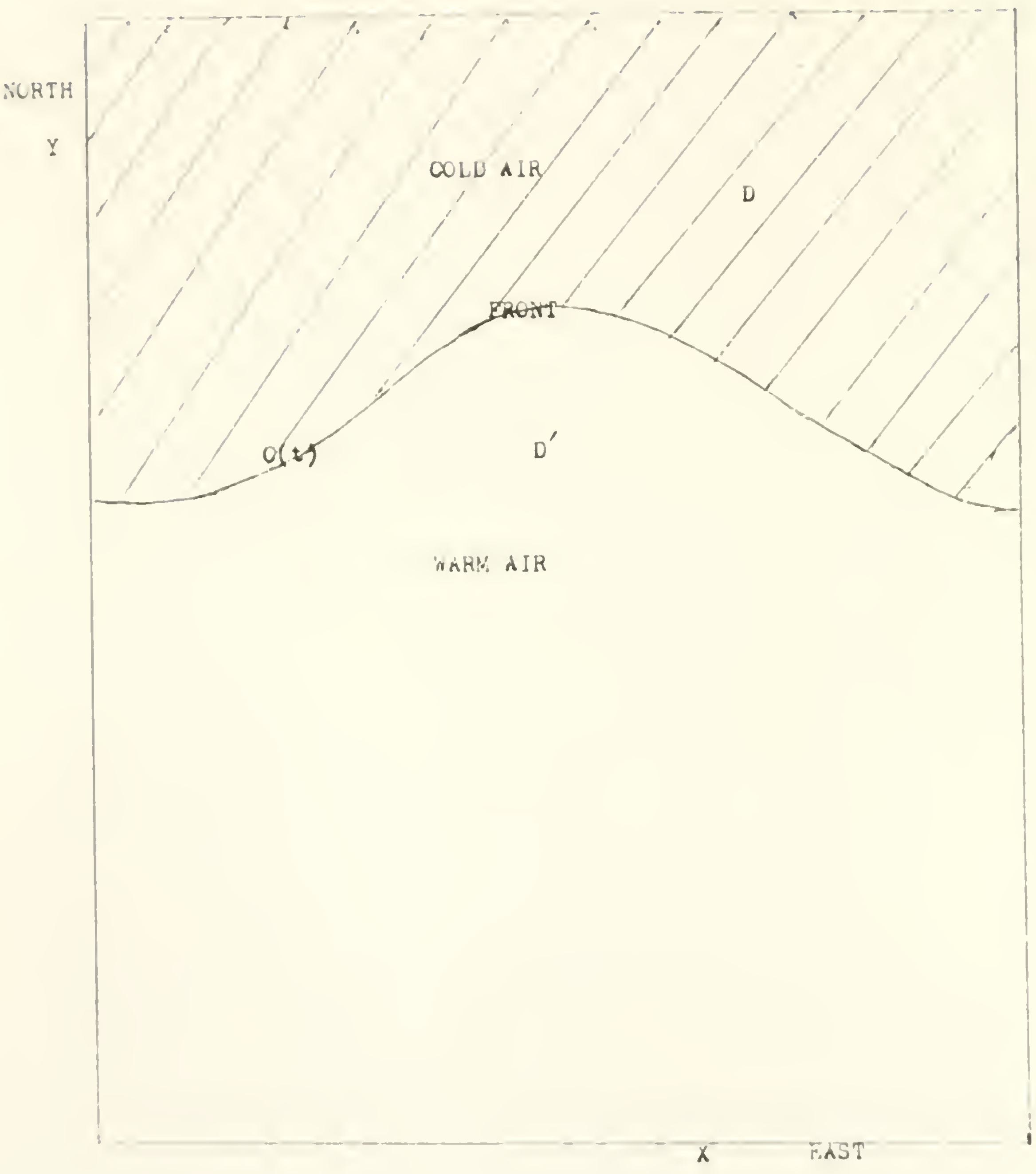
$$\frac{d}{dt}(y(t)) = u_C(x_C, y_C, t)$$

where d/dt is the particle derivative.

We must still prescribe u,v and h along the other three boundaries. For simplicity we shall assume that u,v and h are periodic in the space variable x with a period equal to the distance between the east and west boundaries. Since the atmospheric motion on the earth is periodic in the longitude this condition is correct on a global scale. Furthermore, if occlusion takes place in a relatively small region centered in the period, we may by varying the size of the period determine the influence of the periodicity condition and calculations are made with this object in view. We will also consider one case in which the boundaries are so far apart as to have no influence on the occlusion process and so the domain can be considered as infinite in the x direction.

Along the northern boundary we assume that the normal component of the velocity, i.e. the y component of velocity, v, vanishes for all time. This is in agreement with the assumption made in deriving the one dimensional model, problem IV, so that a meaningful comparison between the two models can be made.

FIGURE 5



A complete formulation of the problem requires appropriate initial data at the time $t = 0$ for u,v,h in the cold air mass domain D. It is also necessary to specify the initial shape of the curve C and the values u,v along the front. The curve C is assumed to be sinusoidal initially. The thickness h of the cold air is assumed to vary linearly in y and to be equal to zero along the front; thus h varies sinusoidally in x. u and v are assumed to be those of a steady state solution of equation (1.9). Since the essential change made from the steady state solution occurs only in the shape of the front our initial conditions can be considered a perturbation of the steady state solution of equation (1.9).

Two different sets of initial conditions are taken, as follows.

Case 1

$$y_C = C_2 - C_1 \cos\left(\frac{2\pi}{X} x\right)$$

$$u = \bar{u}$$

$$v = 0$$

$$h = (y - y_C)H\ , \quad y_C \le y \le Y$$

where $$H = \frac{f}{g\left(1 - \frac{\rho'}{\rho}\right)}\left(\frac{\rho'}{\rho}\bar{u}' - \bar{u}\right)\ , \quad \frac{\rho'}{\rho}\bar{u}' > \bar{u}\ .$$

X, Y denote the lengths of the sides of the rectangle R in the x and y directions respectively and u is the x-velocity

of the warm air. As a second case we assume a physically more relevant condition, i.e. that initially the wind is geostrophic. This condition means that initially there is no acceleration in either the x or y terms. Thus we choose our initial velocities u and v so that $\frac{du}{dt} = 0$ and $\frac{dv}{dt} = 0$.

Case 2

$$h = \left(\frac{y - y_C}{Y - y_C}\right)(Y-b)H$$

$$u = -\frac{g(1-\frac{\rho'}{\rho})}{f}\frac{\partial h}{\partial y} + \frac{\rho'}{\rho}\bar{u}'$$

$$v = \frac{g(1-\frac{\rho'}{\rho})}{f}\frac{\partial h}{\partial x}$$

where $b = C_1 + C_2$, H,Y and y as in Case 1;

with these initial conditions we have at t = 0

$$\frac{du}{dt} = u_t + uu_x + vu_y = -g(1-\frac{\rho'}{\rho})h_x + fv = 0$$

$$\frac{dv}{dt} = v_t + uv_x + vv_y = -g(1-\frac{\rho'}{\rho})h_y - f(u-\frac{\rho'}{\rho}\bar{u}') = 0.$$

Since we have as an initial condition $u = \bar{u}$ the bulge in the front will begin to move to the right and away from the center of the region. We wish to keep this bulge as centered as possible, so that the occlusion pattern will not be affected by the periodicity condition. It is thus convenient to introduce a moving coordinate system so that initially the front is stationary in the x direction. Since the periodicity condition will be imposed in this moving

coordinate system it is possible to thus minimize the effects of the periodicity condition on the occlusion process by forcing the bulge in the front to remain near the center of the domain. At the same time we also introduce dimensionless variables. We therefore introduce the following new variables.

$$
(2.1)\qquad
\begin{aligned}
\tau &= \frac{t}{\Delta t}, & \lambda &= \frac{\Delta t}{\Delta s} \\
\xi &= \frac{x-\bar{u}t}{\Delta s} & \eta &= \frac{y}{\Delta s} \\
\hat{u} &= \lambda(u-\bar{u}) & \hat{v} &= \lambda v \\
\hat{h} &= \lambda^2 g\left(1-\frac{\rho'}{\rho}\right)h & &
\end{aligned}
$$

where $\bar{u}$ is the constant initial velocity, in case I, of the cold air mass. Δt, Δs denote units for time and length respectively. We also introduce the following parameters:

$$
F = f\,\Delta t \qquad\qquad G = F\lambda\left(\frac{\rho'}{\rho}\bar{u}' - \bar{u}\right).
$$

With these new variables the equations become

$$
(2.2)\qquad
\begin{aligned}
&\text{(a)} && \hat{u}_\tau + \hat{u}\hat{u}_\xi + \hat{v}\hat{u}_\eta + \hat{h}_\xi = F\hat{v} \\
&\text{(b)} && \hat{v}_\tau + \hat{u}\hat{v}_\xi + \hat{v}\hat{v}_\eta + \hat{h}_\eta = -F\hat{u} + G \\
&\text{(c)} && \hat{h}_\tau + \hat{h}(\hat{u}_\xi + \hat{v}_\eta) + \hat{u}\hat{h}_\xi + \hat{v}\hat{h}_\eta = 0
\end{aligned}
$$

with initial conditions

<u>Case I</u>

$$
\begin{aligned}
&\text{(d)} && \hat{u}(0,\xi,\eta) = 0 \\
& && \hat{v}(0,\xi,\eta) = 0 \\
& && \hat{h}(0,\xi,\eta) = G(\eta-\eta_C) \quad \text{where } \eta_C = \frac{C_2}{\Delta s} - \frac{C_1}{\Delta s}\cdot\cos\left(\frac{2\pi\,\Delta s}{X}\xi\right).
\end{aligned}
$$

Case II

(d)' $\hat{h}(0,\xi,\eta) = G(\eta - \eta_C)(N-b)/(N-\eta_C)$

$$\hat{u}(0,\xi,\eta) = \frac{1}{F}\left(G - \frac{\partial \hat{h}}{\partial \eta}(0,\xi,\eta)\right)$$

$$\hat{v}(0,\xi,\eta) = \frac{1}{F}\frac{\partial \hat{h}}{\partial \xi}(0,\xi,\eta)$$

where $N = \dfrac{Y}{\Delta s}$, $b = \dfrac{C_1 + C_2}{\Delta s}$ as before,

and

$$\frac{\partial \hat{h}}{\partial \eta} = G\left(\frac{N - b}{N - \eta_C}\right)$$

$$\frac{\partial \hat{h}}{\partial \xi} = -\frac{G(N-b)(N-\eta)}{(N-\eta_C)^2}\frac{\partial \eta_C}{\partial \xi},$$

$$\frac{\partial \eta_C}{\partial \xi} = \frac{2\pi C_1}{X}\sin\left(\frac{2\pi\ \Delta s}{X}\xi\right).$$

Notice that $v(0,\xi,N) = 0$ and so the initial conditions match the boundary conditions at the northern boundary as well as at the other boundaries. The boundary conditions at the northern, eastern and western boundaries are

(e) $\hat{v}(\tau,\xi,N) = 0$

periodicity in the ξ direction.

While the boundary conditions along the front are

(f) $\hat{h}(\tau,\xi,\eta_C) = 0$

$$\frac{dX_C}{d\tau} = \hat{V}_C \qquad \frac{d\hat{V}_C}{d\tau} = \gamma\hat{V}_C - \nabla\hat{h} + K_2$$

where

$$X_C = \begin{pmatrix}\xi_C \\ \eta_C\end{pmatrix} \quad \hat{V}_C = \begin{pmatrix}\hat{u}_C \\ \hat{v}_C\end{pmatrix} \quad \gamma = \begin{pmatrix}0 & F \\ -F & 0\end{pmatrix} \quad K_2 = \begin{pmatrix}0 \\ G\end{pmatrix} \quad \nabla\hat{h} = \begin{pmatrix}\frac{\partial \hat{h}}{\partial \xi} \\ \frac{\partial \hat{h}}{\partial \eta}\end{pmatrix}$$

For the rest of this chapter we will be dealing only with this dimensionless moving coordinate system unless otherwise mentioned. Thus, from now on we shall omit writing the circumflex for dimensionless variables and we shall use x,y,t for the moving coordinate system instead of ξ, η, τ.

b. Difference Equations

We consider the rectangle R with sides of length L_1, L_2. We choose a rectangular mesh in such a way that the coordinates of the grid are defined by

$$X_i = \frac{iL_1}{I\ \Delta s} \qquad i = 0,1,\ldots,I$$

$$Y_i = \frac{jL_2}{J\ \Delta s} \qquad j = 0,1,\ldots,J$$

where the boundary lines correspond to $i = 0,I$; $j = 0,J$. For the unrefined mesh we choose $I = \frac{L_1}{\Delta s}$, $J = \frac{L_2}{\Delta s}$ so that X, Y are integers.

We define D_Δ as the connected set of net points in the interior of D (i.e. we exclude the points at the northern boundary and points on the front). By a regular point we mean a point in D_Δ whose eight nearest neighbors are all in D_Δ, all other points in D_Δ are called irregular points. At regular points we consider two different second order schemes. The first is a one-step scheme that is a generalization of the Lax-Wendroff scheme [12] to nonlinear equations. The second method is a two-step scheme due to Burstein [3].

In order to formulate the one-step method we write the differential equation in vector form. Let W be the vector of dependent variables and $A = (a_{ij})$, $B = (b_{ij})$, and $C = (c_{ij})$ be matrices that could depend on W.

$$W_t = AM_x + BW_y + C$$

so

$$W_{tt} = A_t W_x + AM_{xt} + B_t W_y + BW_{yt} + C_t$$

where

$$W_{xt} = A_x W_x + AW_{xx} + B_x W_y + BW_{yy} + C_y$$

$$A_t = (a_{ij,t}), \quad B_t = (b_{ij,t}), \quad C_t = (c_{ij,t}) \ .$$

Thus, W_{tt} is given in terms of space derivatives only. We then use these time derivatives to calculate W at the new time step by using Taylor series

$$W(t+\Delta t) = W(t) + (\Delta t)W_t(t) + (\Delta t)^2 W_{tt}(t) + O((\Delta t)^3)$$

For the finite difference scheme we ignore all terms of order $(\Delta t)^3$ and replace all space derivatives by centered differences.

Thus, let

$$T_x^s f = f(x + s\ \Delta x,\ y) \qquad T_y^s f = f(x, y + s\ \Delta y)$$

Then

$$\frac{\partial}{\partial x} \rightarrow \frac{1}{2\Delta x}(T_x - T_x^{-1}) \qquad \frac{\partial}{\partial y} \rightarrow \frac{1}{2\Delta y}(T_y - T_y^{-1})$$

$$\frac{\partial^2}{\partial x^2} \rightarrow \frac{1}{(\Delta x)^2}(T_x - 2I + T_x^{-1}) \qquad \frac{\partial^2}{\partial y^2} \rightarrow \frac{1}{(\Delta y)^2}(T_y - 2I + T_y^{-1})$$

$$\frac{\partial^2}{\partial x \partial y} \rightarrow \frac{1}{r(\Delta x)(\Delta y)}(T_x - T_x^{-1})(T_y - T_y^{-1})$$

Let $\lambda = \max\left(\frac{\Delta t}{\Delta x}, \frac{\Delta t}{\Delta y}\right)$. If A, B and C were constant and if we were to consider the pure initial value problem then this scheme would be stable if $\lambda \leq \frac{1}{\sigma\sqrt{8}}$, where σ is the largest eigenvalue of either A or B (in terms of absolute value) [13]. It turns out empirically (see for example reference [2]) that for the fluid dynamic equations this condition is too stringent and can be exceeded in practice without causing instability. In particular it was found that the value of λ could be doubled without causing instability. However, in using this criterion it must be remembered that near the front the distance between the front points and the mesh points can be considerably less than Δx. It thus seems reasonable to modify the difference scheme at points close to the front. The eigenvalues of A and B are $u+c$, u, $u-c$; $v+c$, v, $v+c$. Thus we require that $\Delta t \leq \frac{K\ \Delta s}{\max(|u|,|v|) + \sqrt{h}}$ where Δs is the distance between the points used in the calculation and K is a constant to be determined by trial and error and is approximately equal to $\frac{1}{\sqrt{2}}$. Near the front where Δs is small h is also small, so that there is some compensation for the short distances involved.

For the two-step method at the regular points we convert the system of differential equations to conservative form. A partial differential equation is said to be in conservative form if it can be written as

$$w_t + f_x + g_y = r \qquad \text{where } f = f(w,x,y,t),\ g = g(w,x,y,t).$$

With our system this can be accomplished by letting

$$w = \begin{bmatrix} hu \\ hv \\ h \end{bmatrix} \quad f = \begin{bmatrix} hu^2 + \frac{1}{2} h^2 \\ huv \\ hu \end{bmatrix} \quad g = \begin{bmatrix} huv \\ hv^2 + \frac{1}{2} h^2 \\ hv \end{bmatrix} \quad r = \begin{bmatrix} Fu \\ -Fv+G \\ 0 \end{bmatrix} .$$

A two-step Lax-Wendroff scheme is a second order scheme in which temporary values are generated by a first order scheme. Then, in a second step this intermediate data is used to generate the values of the variables at the next time level and these values are accurate up to terms of order $((\Delta t)^3)$. A general form for a second order scheme is

Step 1: $\tilde{w}(t+nk) = P_n w(t)$

Step 2: $w(t+k) = Q_n w(t) + R_n \tilde{w}(t+nk)$.

These two steps were originated by Richtmyer as generalizations of the modified Euler method for ordinary differential equations. However his scheme separates those points where i+j are even from those where i+j are odd and it has been found (see for example reference 7) that the Richtmyer scheme can be weakly unstable; i.e. there can be a divergence between the $2\Delta x$ and $4\Delta x$ components of the solution because of this lack of coupling between neighboring points. We shall therefore use a two-step scheme proposed by Burstein [3]. In this case n = 1 in our general formula and so our scheme simulates a predictor-corrector scheme but with only one correction.

$$w(x_{i+1/2}, y_{j+1/2}, t+\Delta t) = \frac{1}{4}(w_{i,j} + w_{i+1,j} + w_{i,j+1} + w_{i+1,j+1})$$

$$- \frac{\Delta t}{2\Delta x}\{f(w_{i+1,j}) - f(w_{i,j}) + f(w_{i+1,j+1}) - f(w_{i,j+1})\}$$

$$- \frac{\Delta t}{2\Delta y}\{g(w_{i+1,j+1}) - g(w_{i+1,j}) + g(w_{i,j+1}) - g(w_{i,j})\}$$

$$+ \frac{\Delta t}{4}\{r(w_{i+1,j+1}) + r(w_{i+1,j-1}) + r(w_{i-1,j+1}) + r(w_{i-1,j-1})\}$$

$$ww(x_i, y_j, t+\Delta t) = w(x_i, y_j, t)$$

$$- \frac{\Delta t}{4\Delta x}\{f(w_{i+1,j}) - f(w_{i-1,j}) + f(\tilde{w}_{i+1/2,j+1/2})$$

$$- f(\tilde{w}_{i-1/2,j+1/2}) + f(\tilde{w}_{i+1/2,j-1/2}) - f(\tilde{w}_{i-1/2,j-1/2})\}$$

$$- \frac{\Delta t}{4\Delta y}\{g(w_{i,j+1}) - g(w_{i,j-1}) + g(\tilde{w}_{i+1/2,j+1/2})$$

$$- g(\tilde{w}_{i+1/2,j-1/2}) + g(\tilde{w}_{i-1/2,j+1/2}) - g(\tilde{w}_{i-1/2,j-1/2})\}$$

$$+ \frac{\Delta t}{2}\{r(w_{i,j}) + r_{ij}(t + \Delta t)\},$$

where

$$r_{ij}(t + \Delta t) = \frac{1}{4}\{r(\tilde{w}_{i+1/2,j+1/2}) + r(\tilde{w}_{i+1/2,j-1/2})$$

$$+ r(\tilde{w}_{i-1/2,j+1/2}) + r(\tilde{w}_{i-1/2,j-1/2})\}$$

The amplification matrix corresponding to the linearized equation is

$$H(\xi,\eta) = I + i\lambda\left\{A \sin\xi\left(\frac{3+4\cos\eta}{4}\right) + B\sin\eta\left(\frac{3+4\cos\xi}{4}\right)\right\}$$

$$+ \frac{\lambda^2}{2}\left\{A[(1-\cos\xi)(1+\cos\eta)]^{1/2} + B[(1-\cos\eta)(1+\cos\xi)]^{1/2}\right\}^{1/2}$$

The eigenvalues of this matrix can be computed numerically and it is found that the stability requirement is $\frac{\Delta t}{\Delta x} \leq \frac{.7550}{\sigma}$ where σ is the maximum sound speed, in our case $c = \sqrt{u^2+v^2} + \sqrt{h}$.

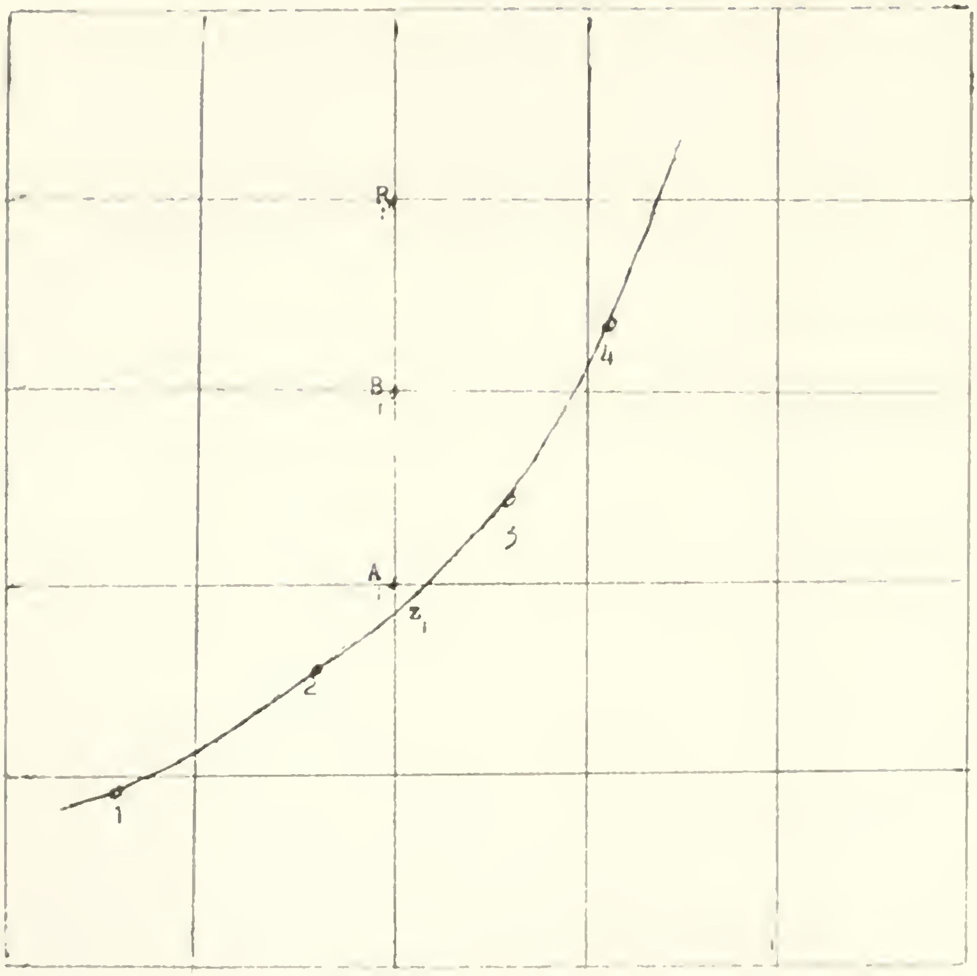
This method has the advantage that it doesn't involve matrix multiplication and so is much faster and it also allows a slightly larger value for Δt. However, it was found that much of the machine time was spent on calculations having to do with points on the front and so the time saved in advancing the solution at regular net points is not large compared to the total time required for the solution. The disadvantages of this method are that it is not a dissipative scheme and that it does not generalize to the irregular points where all eight neighbors are not in the domain of interest.

We now consider ways of achieving second order accuracy at the irregular points. Following a procedure due to Richtmyer [14] we divide the irregular mesh points into two classes. A type "A" point is an irregular point lying closer to the boundary than a certain distance δ (taken to be one quarter of the grid distance). A type "B" point is an

irregular point not of type "A". At type "A" points the differential equations are not used since Gary [14] has found instabilities if this is not done. The reason for this is that if the net point is very close to the front then a small error in the values of the variables on the front or at the mesh points, will cause large oscillations in the approximating polynomial based on these values and so there will be large inaccuracies in the evaluation of the derivatives. According to the physical interpretation of the stability criteria for hyperbolic equations, based on the theory of Courant, Friedrichs and Lewy [5] one would expect trouble whenever the distance used in the computation is appreciably less thatn $c\ \Delta t$. Thus, if we would use the difference equations for points very near to the front we would have to reduce Δt to an unreasonably small value.

To avoid this difficulty we evaluate u,v and h at type "A" points by interpolation, instead of using the finite difference equations. We first find u, v and h at all regular points, at type "B" points and at the points on the front (by methods to be described later). In the usual case we first interpolate using front points 1, 2, 3, 4 (see Figure 4) to find u and v on the front along the same x-coordinate line as A_1. We use these values together with those at the mesh points B_1, R_1 (but not other type "A" points) on the same x-coordinate line as A_1 to find u,v,h at A_1 with second order accuracy. When the slope of the front is very large we reverse the situation and do the

FIGURE 4

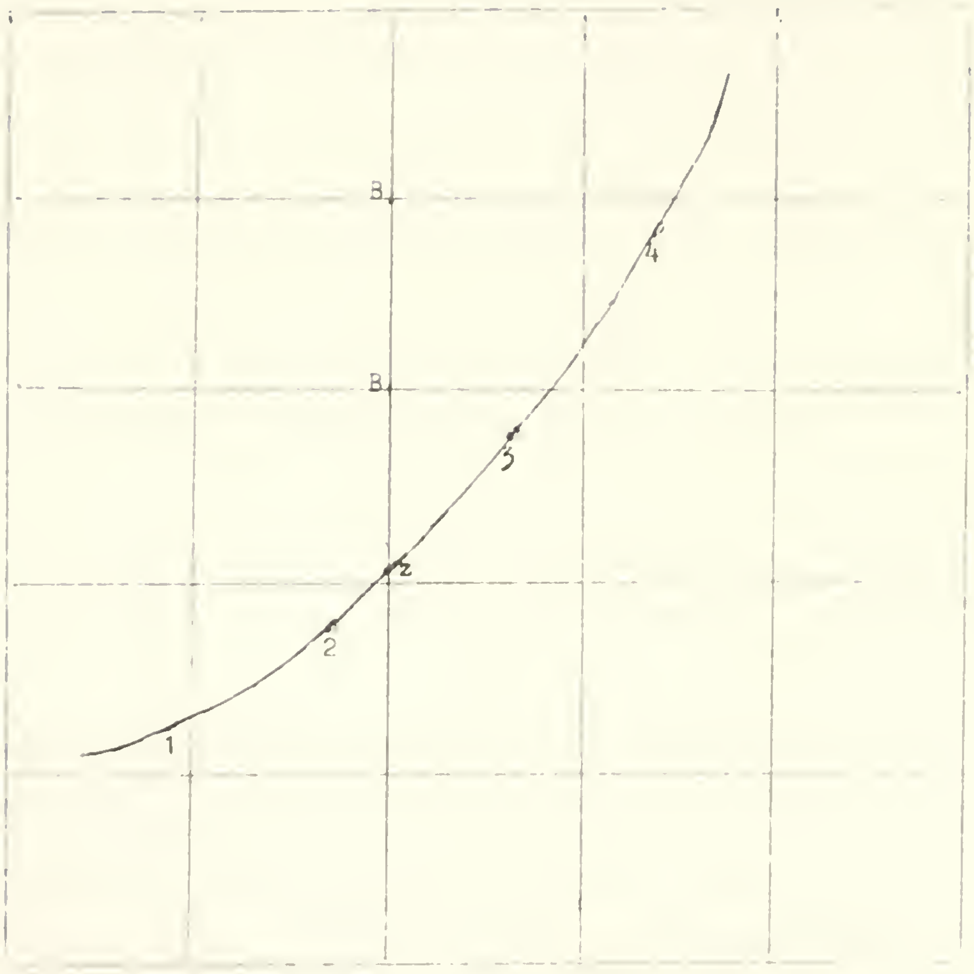


u at A_1 is obtained by interpolation using the values at z_1, B_1, and R_1. The value at z_1 is gotten by using the values at the front points 1,2,3,4.

interpolation along a horizontal line. In all cases encountered here it was possible to interpolate in either a horizontal or a vertical direction.

At type "B" points we use the one-step Lax-Wendroff scheme independent of which scheme was used at the regular points. The only change is that now we must use noncentered formulas to achieve second order accuracy in evaluating the space derivatives. This occurs because the mesh points used in the calculations are no longer equidistant. Thus to find u at the point B_2 (see Figure 5) we first find the location of the point z_2 on the front along the same vertical line as B_2. Generally this was done by fitting a cubic curve to the front points. However, when the slope of the front changes radically, as when the occlusion process begins, it was found that a cubic polynomial oscillated too much and a simple linear fit gave better accuracy. The use of a linear approximation does not affect the order of accuracy since it is used at only a few points; furthermore, since the slope is so steep the front is nearly a straight line. In all these cases the same number of points were used on either side of the coordinate line to prevent distortions. Thus, at most of the points a cubic polynomial was used even though a quadratic would have sufficed for second order accuracy. Once we have found the position of z_2 we find the values of u and v at z_2 by interpolating along the front. We then evaluate u_y at B_2 by using the values of u at R_2, B_2, and z_2.

FIGURE 2



u is obtained at B by using an uncentered difference scheme involving R ,B ,z . u at z is gotten by interpolation along the front using points 1,2,3,4.

A similar process is used for the x derivatives and for all the second order derivatives. Thus, if k is the distance between B_2 and z_2 we have the following formulas.

$$u_y\Big|_{B_2} = \frac{k}{(\Delta y)(k+\Delta y)}\, u(R_2) + \frac{\Delta y - k}{k\ \Delta y}\, u(B_2) - \frac{\Delta y}{k(k+\Delta y)}\, u(Z_2)$$

$$(2.3)\qquad u_{yy}\Big|_{B_2} = \frac{2}{(\Delta y)(k+\Delta y)}\, u(R_2) - \frac{2}{k\ \Delta y}\, u(B_2) + \frac{2}{k(k+\Delta y)}\, u(Z_2)$$

$$u_{xy}\Big|_{B_2} = \left(\frac{u(A_1) - u(R_3)}{2\ \Delta x} - u_x\Big|_{B_2}\right) / \Delta y\ .$$

c. Boundary Conditions

We next consider the finite difference approximations at all the boundaries. Along the east and west boundaries we use the regular difference approximations using the periodicity condition to obtain the values of the dependent variables at the neighboring points. Along the northern boundary we have v = 0 for all time. To find the values of u and h we use one sided difference approximations and arrive at formulas similar to those in (2.3).

It remains to satisfy the boundary conditions along the front as given by equations (2.2f). Along the front h = 0 by definition and so we must solve the differential equations for u and v. To solve this system of first order ordinary differential equations two different methods were tried. The first method was the trapezoidal rule, which is an implicit method.

$$(2.4)\quad (a)\quad X_\ell(t+\Delta t) = X_\ell(t) + (\Delta t)\langle V_\ell\rangle$$

$$(b)\quad V_\ell(t+\Delta t) = V_\ell(t) + (\Delta t)[\gamma\langle V\rangle - \langle\nabla h_\ell\rangle + K_2]$$

Here the symbol $\langle\ \rangle$ denotes the averaging operator $\langle w\rangle = \frac{1}{2}(w(t) + w(t+\Delta t))$, and γ and K_2 are the same as in equation (2.2f). Following K.I.S., we solve (2.4b) for $V_\ell(t+\Delta t)$ with the result

$$(2.4)\quad (c)\quad V_\ell(t+\Delta t) = \alpha\, V_\ell(t) - \beta\langle\nabla h_\ell\rangle + \beta\, K_2$$

where

$$\alpha = \frac{1}{1+(\frac{F}{2}\Delta t)^2}\begin{bmatrix} 1-(\frac{F\,\Delta t}{2})^2 & F\,\Delta t \\ -F\,\Delta t & 1-(\frac{F\,\Delta t}{2})^2 \end{bmatrix}$$

$$\beta = \frac{1}{1+(\frac{F}{2}\Delta t)^2}\begin{bmatrix} \Delta t & \frac{F(\Delta t)^2}{2} \\ -\frac{F(\Delta t)^2}{w} & \Delta t \end{bmatrix}$$

To solve for the dependent variables with the use of this implicit scheme the following procedure was used:

(a) Set $\nabla h_\ell(t+\Delta t) = \nabla h_\ell(t)$

(b) Predict $V_\ell(t+\Delta t)$ using (3.3c)

(c) Move the front using (3.3a)

(d) Calculate new values at the type "A" points

(e) Calculate $\nabla h_\ell(t+\Delta t)$ by a method to be described

(f) If there is any significant change in $V_\ell(t+\Delta t)$ from the previous iterate then the process is repeated starting with step (b).

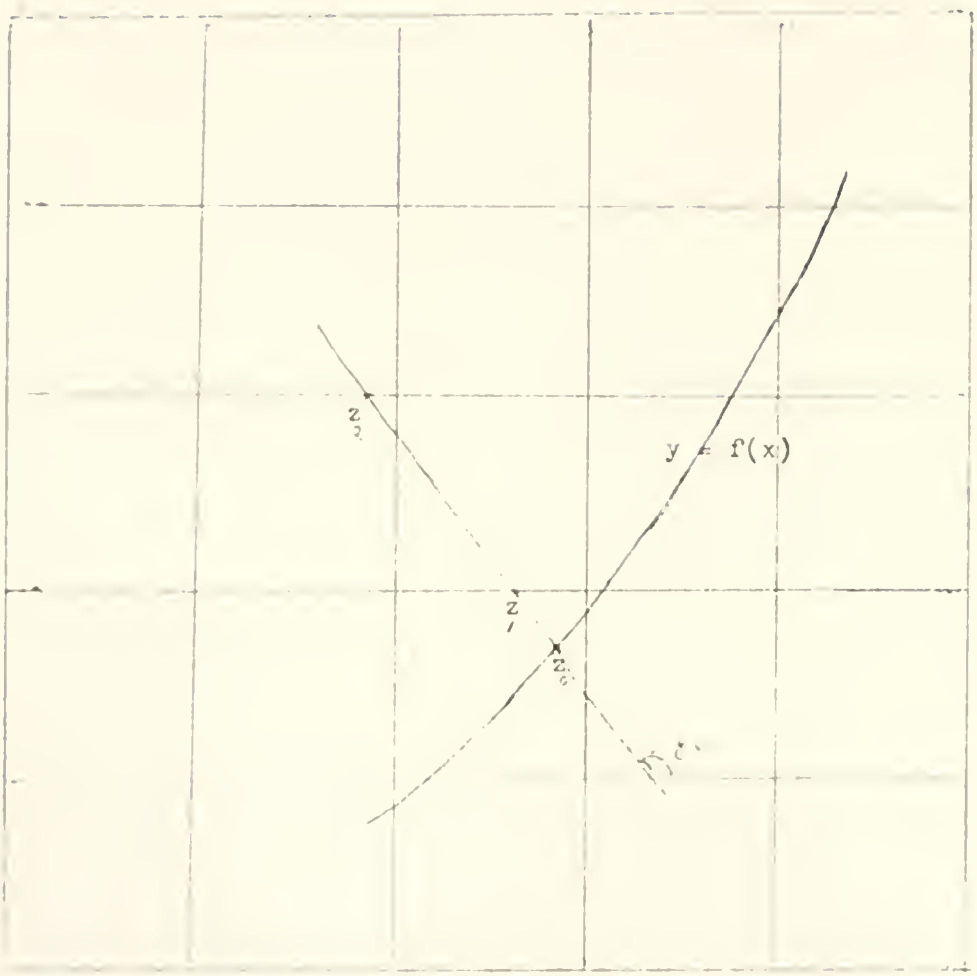
Another second order method for solving ordinary differential equations is the Adams-Bashforth method.

$$
\begin{aligned}
&X_\ell(t+\Delta t) = X_\ell(t) + \frac{\Delta t}{2}[3V_\ell(t) - V_\ell(t-\Delta t)] \\
(2.5)\quad &V(t+\Delta t) = V_\ell(t) + \frac{\Delta t}{2}[\gamma(3V_\ell(t) - V_\ell(t-\Delta t) \\
&\qquad - (3\nabla h_\ell(t) - \nabla h_\ell(t-\Delta t)) + K_2] \; .
\end{aligned}
$$

This method has the advantage that it is explicit and that it is unconditionally stable while the trapezoidal rule requires iterations and is weakly unstable. However, the Adams-Bashforth method requires knowledge of the values of the variables at time $t-\Delta t$. At time Δt we can avoid this difficulty by finding the solution by a finite Taylor series instead of the finite difference scheme; however, when we redistribute the points along the front we no longer know the values of u, v and gradient h at the new points for previous times. It was found that until the time that the front points were redistributed for the first time both methods gave similar results and so it was decided to use the implicit method only.

Both methods for integrating the ordinary differential equations require the evaluation of the gradient of h at the front. To accomplish this we first evaluate the normal derivative of h at the front. The slope of this normal is $-1/\frac{dy_C}{dx}$, where $\frac{dy_C}{dx}$ is found by using a quadratic fit along the front and then differentiating this polynomial. We draw a straight line from a front point with the slope of the normal into the cold air mass (see Figure 6). We find where

FIGURE 6



The values of h at z_1, z_2 are calculated using the values at the circled points. These values are then used to find $\frac{\partial h}{\partial n}(\zeta)$.

this line crosses the first two horizontal coordinate lines that it meets, i.e. at z_1, z_2. If the first coordinate line is less than $\delta = \frac{\Delta x}{4}$ away from the front then we skip that coordinate line and consider the next two. We do this check for similar reasons to our not using the difference equations at type "A" points, i.e. in order to avoid instabilities caused by oscillations in the approximating polynomials. We then find the values of h at z_1 and z_2 by quadratic interpolation along a y-coordinate axis. If the slope of the front is too large we again switch the roles of the x and y axes and find the value of h at the intersection of the normal and the first two x-coordinate lines. Since h = 0 along the front we know the value of h at three distinct points and so we can find the value of the derivative of h along the normal to the front with second order accuracy. Also, since h is identically zero along the front, by definition, the tangential derivative along the front is zero. We thus know both the normal and tangential derivatives of h and so we can find all the directional derivatives of h at the front and in particular the gradient of h. In fact we have

$$h_x = \pm \frac{\partial h}{\partial n} \Big/ \sqrt{1 + \left(\frac{dy_C}{dx}\right)^2} \,, \quad h_y = \pm \frac{\partial h}{\partial n} \frac{dy_C}{dx} \Big/ \sqrt{1 + \left(\frac{dy_C}{dx}\right)^2}$$

The sign in these formulas is determined by the direction of the normal into the cold air.

2. Redistribution of the Front Points

It was found that as the occlusion process continued the individual particles on the front converged toward the center of the bulge and so the spacing between the front points becomes small in comparison with the grid size. This leads to an inaccurate exchange of information between the front and the mesh points. In addition the uneven spacing of the front point caused oscillations in the approximating polynomial to the front. It was thus found necessary to redistribute the front points from time to time. In order to redistribute the particles along the front at time t we calculated the arclength between particles. This was done by passing a quadratic curve $y_C = \frac{\alpha}{2} x^2 + \beta x + \gamma$ through the points i-1, i, i+1. Then

$$\frac{dy_C}{dx} = \alpha x + \beta \ , \qquad \left(\frac{dy_C}{dx}\right)^2 + 1 = ax^2 + bx + c$$

and

$$s_i - s_{i-1} = \int_{x_{i-1}}^{x_i} \sqrt{1+\left(\frac{dy_C}{dx}\right)^2}\, dx = \frac{1}{4\alpha}\left\{(2\alpha x + \beta)\sqrt{ax^2+bx+c} + \log\left|2\alpha x + \beta + \sqrt{ax^2+bx+c}\right|\right\}_{x_{i-1}}^{x_i} .$$

When the slope of the front became very steep we considered x as a function of y and arrived at similar formulas.

We then distributed the points along the front in such a manner that the points were equally spaced in terms of arclength. We found the new values of x, y, u and v

by interpolation considering these variables as functions of arclength. Thus everything is determined within an arbitrary constant which can be fixed by specifying that $s = 0$ at $x = 0$.

e. Data and Results of the Calculations

The following numerical values for the parameters in the problem were taken.

$$\Delta s = 250{,}000 \text{ feet} \simeq 50 \text{ miles}$$

$$\Delta t = 1{,}800 \text{ seconds} \simeq \frac{1}{2} \text{ hour}$$

$$\lambda = \frac{\Delta t}{\Delta s} = .0072 \text{ second/feet}$$

$$X = 20\ \Delta s \text{ or about } 1{,}000 \text{ miles}$$

$$Y = 20\ \Delta s$$

$$C_1 = 9.5\ \Delta s$$

$$C_2 = 2\ \Delta s$$

$$g = 32.1521 \text{ feet/second}$$

$$f = 10^{-4}/\text{second}$$

$$\bar{u} = 10 \text{ feet/second}$$

$$\frac{\rho'}{\rho} = 1 - \frac{3}{5g} \simeq .982$$

$$\bar{u}' = 50 \frac{\rho'}{\rho} \text{ feet/second}$$

So

$$F = f\ \Delta t = .18$$

$$G = F\lambda\left(\frac{\rho'}{\rho}\ \bar{u}' - \bar{u}\right) = .05184$$

$$\hat{h} = 3.1104 \times 10^{-5}\ h$$

and in case I we have

$$\left.\frac{\partial h}{\partial y}\right|_{t=0} = \frac{f}{g\left(1 - \frac{\rho'}{\rho}\right)} \left(\frac{\rho'}{\rho}\ \bar{u}' - \bar{u}\right) \simeq \frac{1}{150}\ .$$

The above magnitude of the density discontinuity corresponds to a temperature discontinuity of about 5° centigrade. The initial slope of the stationary discontinuity is in the range of observed values for the slope of frontal surfaces in the middle latitudes. The time step was chosen as $\Delta t = 1/3$ on the basis of the stability criterion discussed in the previous section. This time step was reduced by three-fourths after nine hours because of the increase in the magnitude of the velocity in the domain D.

We first describe the result for case I where the initial velocities are zero in the moving coordinate system. Figure 7 shows the initial position of the front, which separates the domain of the cold air from that of the warm air. The mesh sizes, in the unrefined mesh, were taken as having length and width 1. For convenience in programming two extra columns of grid points were added outside the western and eastern boundaries in order to satisfy the periodicity condition. All the diagrams in this section will refer to the moving coordinate system except where expressly noted otherwise. Figure 8a shows the position of the front after 8 hours. This result agrees graphically with that of K.I.C. The following figures show the position of the front at one hour intervals until a total of 16 hours has elapsed. Evidently, the entire front progresses to the east relative to the moving coordinate system. The cold front moves eastward faster than the warm front. Thus, the

bulge of the warm air into the cold air narrows. We notice that the front is no longer a single-valued function of the x coordinate and that the front is beginning to curl counterclockwise. The development of this asymmetry suggests the occlusion process of frontal cyclones which agrees with the qualitative analysis given by Whitham [19] and Stoker [16]. We also observe that after approximately 14 hours a second front forms to the west of the original front. At this new front the warm air does not bulge into the cold air as far as in the original front. Between these two fronts the depth of the cold air is quite small. Thus, a short distance above the earth's surface these two fronts merge together.

In order to show the movement of the front in detail the trajectories of individual points on the front during the first eight hours is shown in Figure 9. By later times the points on the front have been redistributed several times and so it would be difficult to follow the trajectory of individual points. We note that points on the cold front move southeastward and those on the warm front move northeastward on the average whereas both fronts propagate eastward. The movement of the front clearly indicates the production of a cyclonic circulation about the circulation center where the cold and warm fronts meet. In this figure we also show the magnitude of the velocity components. The numerators are the x-component of the velocity while the denominators are the y-component of the velocity. Note the sharp change

in the components of the velocity near the circulation center. To illustrate the circulation pattern even more clearly we show a plot of the velocity field, in the original coordinate system, in Figure 10. In these plots the direction of the arrow shows the direction of the velocity field while the length of the arrow is proportional to the magnitude of the velocity. In Figure 11 we show contour lines of the height of the cold air mass at 5,000 foot intervals. In this graph $Y = 26\ \Delta s$ so that we include more contour lines and so that this corresponds to the graph of K.I.S.

The trajectories of the points on the front near the circulation center shows the converging motion of the cold air (see Figure 9). Consequently, the spacing between these points on the front becomes small compared with the grid spacing. This uneven spacing introduces large errors in all the interpolation formulas. Therefore, it was necessary to redistribute the points on the front to equalize the spacing. However, it was found that if the redistribution was done too often the results were smoothed out too much and the deformation of the front no longer resembled the occlusion process. Thus, the points on the front were redistributed after 6 and 8 hours and then every hour afterwards. Because of the high curvature near the center of circulation it was necessary to approximate the front there by a linear function rather than a quadratic or cubic function. The higher order polynomials had excessive

oscillations because of this large curvature. For the same reason the velocity components near the circulation center were calculated by linear interpolation rather than using higher order polynomials.

This program was then repeated for the southern hemisphere. Here $f = \omega \sin \phi$ is now negative since ϕ is negative. However, as initial conditions we take the x component of the cold and warm air velocities as negative. Thus, initially both layers are moving westward instead of eastward. As before the y component of the velocity is zero in both layers. When we introduce the moving coordinate system it is now moving westward instead of eastward. In this moving coordinate system we have

$$F_s = f_s \Delta t = - F_N , \quad G_s = F_s \frac{\Delta t}{\Delta s} \left(\frac{\rho'}{\rho} \bar{u}_s' - \bar{u}_s\right) = G_N .$$

As expected the front follows a similar paytern except that the occluded front moves to the west as seen in Figure 12.

As a check on the program and also to improve the accuracy the program was repeated with a finer mesh. This can be done in two ways. We can keep the same dimensionless moving coordinate system described in the beginning of this chapter and choose the length and width of the mesh sizes as $\frac{1}{2}$ their original value. Then according to the stability criterion we must halve the time step so that $\Delta t = 1/6$. An alternative method is to change the coordinate system so that $\Delta s = 125{,}000$ feet, $\Delta t = 900$ seconds and $\lambda = .0072$ sec./ft. and keep $\Delta x = \Delta y = 1$. Both methods were tried and gave

the same result as they obviously should. After 8 hours we have results essentially identical with those obtained using the coarser mesh (see Figure 13). However, as the curvature near the circulation center increases the additional front points give greater accuracy. Similarly, in the cold air near the circulation center there are large gradients in the height which are measured with greater accuracy by the finer mesh.

To make the model more realistic a third case was introduced. In case III the initial conditions were changed so that the front is sinusoidal only in the middle of the region of numerical integration. Near the east and west boundaries the front is now a straight line parallel to the northern boundary. By making these straight portions long enough we can eliminate the effect of the eastern and western boundaries on the occlusion process for the times considered. Thus we are simulating an infinite domain in the x-direction and so we can ignore the periodicity condition which was artificially introduced for mathematical convenience. We thus have

Case III

$$y_C(x) = \begin{cases} C_2 - C_1 & 0 \leq x \leq K \\ C_2 - C_1 \cos\left(\frac{2\pi\ \Delta s}{X}\ x\right) & K \leq x \leq K+X \\ C_2 - C_1 & K+X \leq x \leq 2K+X \end{cases}$$

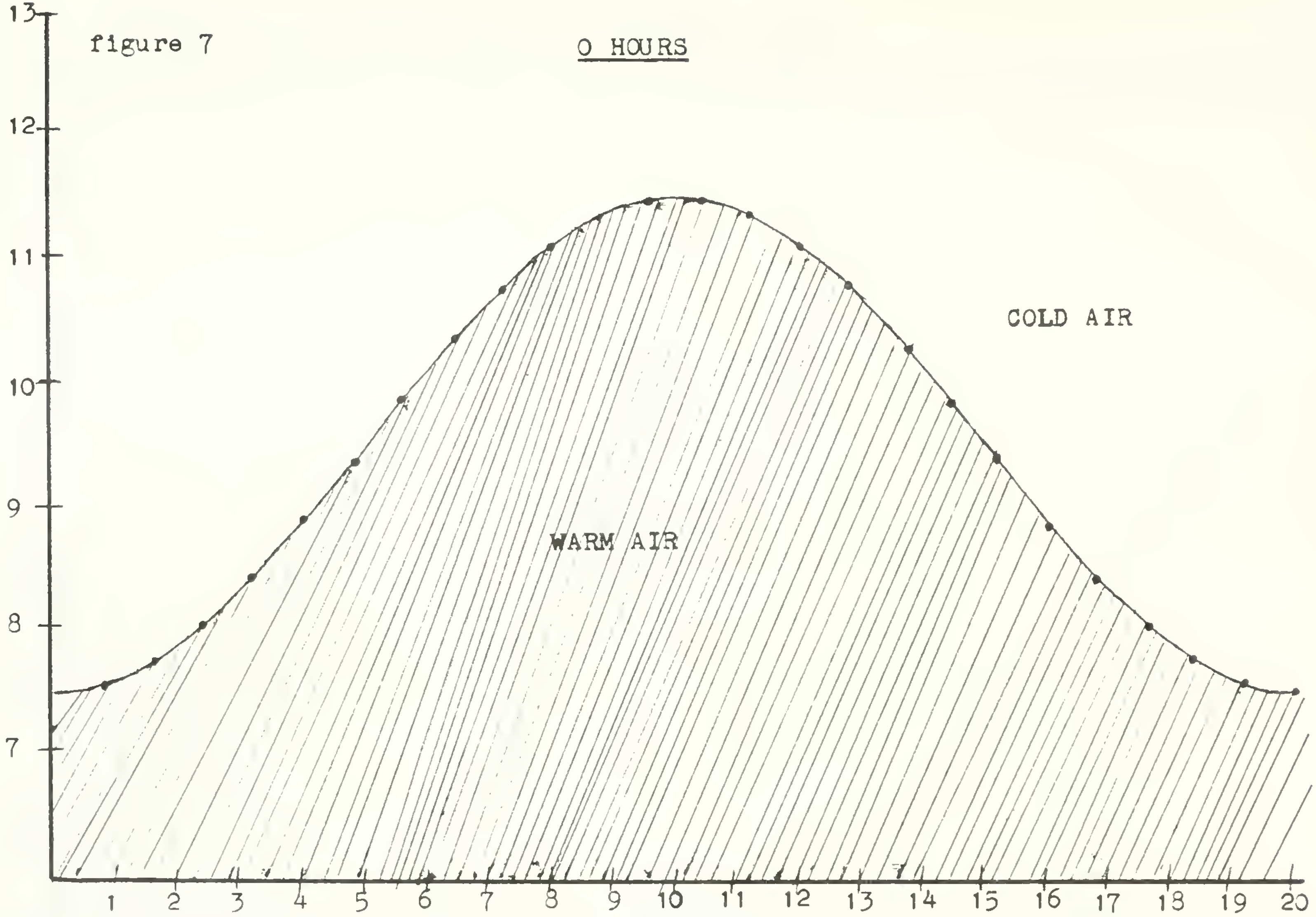
The rest is the same as in case I in the enlarged domain $0 \leq x \leq 2K+X$. In the graphs shown here we used $K = X = 20$.

As seen in Figure 14 the occlusion process is unchanged by the new boundary conditions. However, previously the front had been forced northward near the eastern and western boundaries by the periodicity condition. Now the front remains closer to its initial value and even moves somewhat southward to the west of the front. This movement increases the length of the cold front where we have the steep slope.

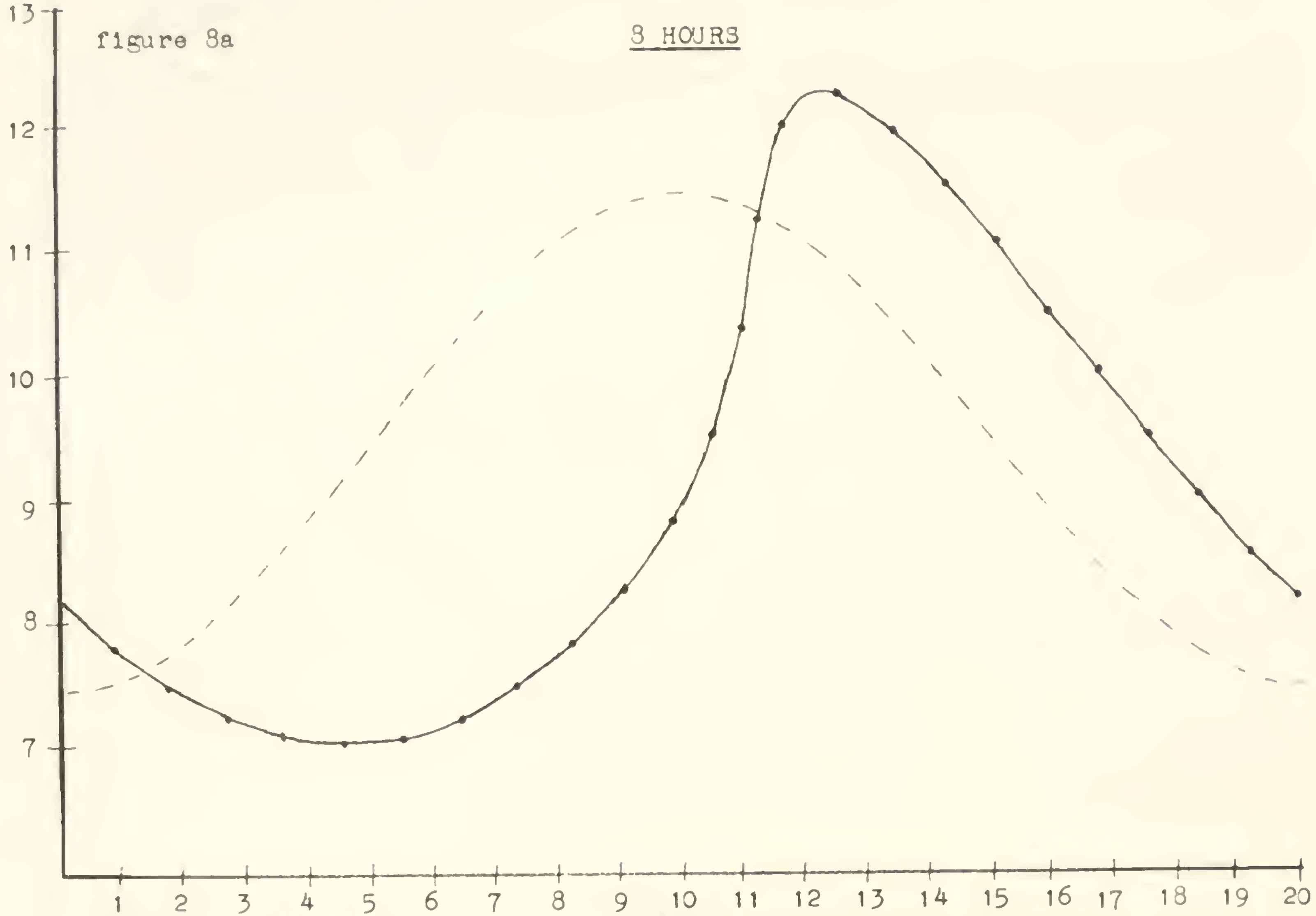
In case II we change the initial velocities and height distribution to conform to the condition of a geostrophic wind. The position of the front is again initially sinusoidal and since the initial wind field is geostrophic we have $\frac{du}{dt} = 0$ and $\frac{dv}{dt} = 0$. Thus, the initial slope of the height of the cold air, $\frac{\partial h}{\partial y}$, is constant with respect to y but is variable with respect to x. The boundary conditions are the same as in our original formulation, case I. These initial conditions are physically more reasonable than those of initially constant velocities. However, this prolem was more difficult to handle numerically. Figure 15 shows the position of the front after 11 hours and 18 hours have elapsed. As in the previous cases the entire frontal system progresses eastward relative to the moving coordinate system. The cold front moves faster than the warm front and we again observe the beginnings of the occlusion process. As was done previously this program was repeated using the finer mesh

with substantially the same results as seen in Figure 16. In this case our calculations do not completely agree with Kasahara and the occlusion process is not observed until much later times than is noted by Kasahara.

figure 7



Initial position of the front for cases I and II.
The points on the front are the material particles on the frontal surface whose motion determines the position of the front.



Position of the front for case I after 8 hours.

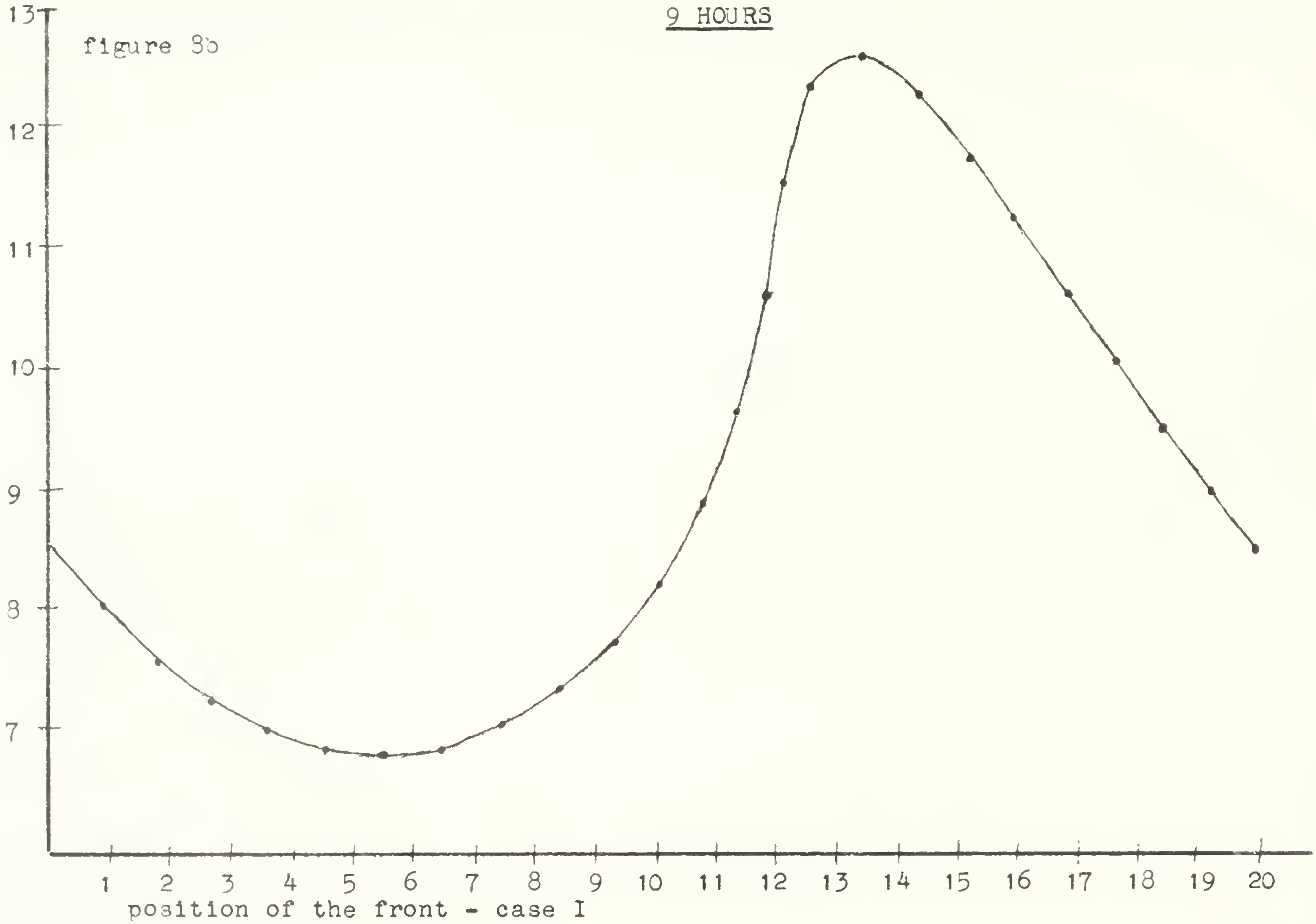
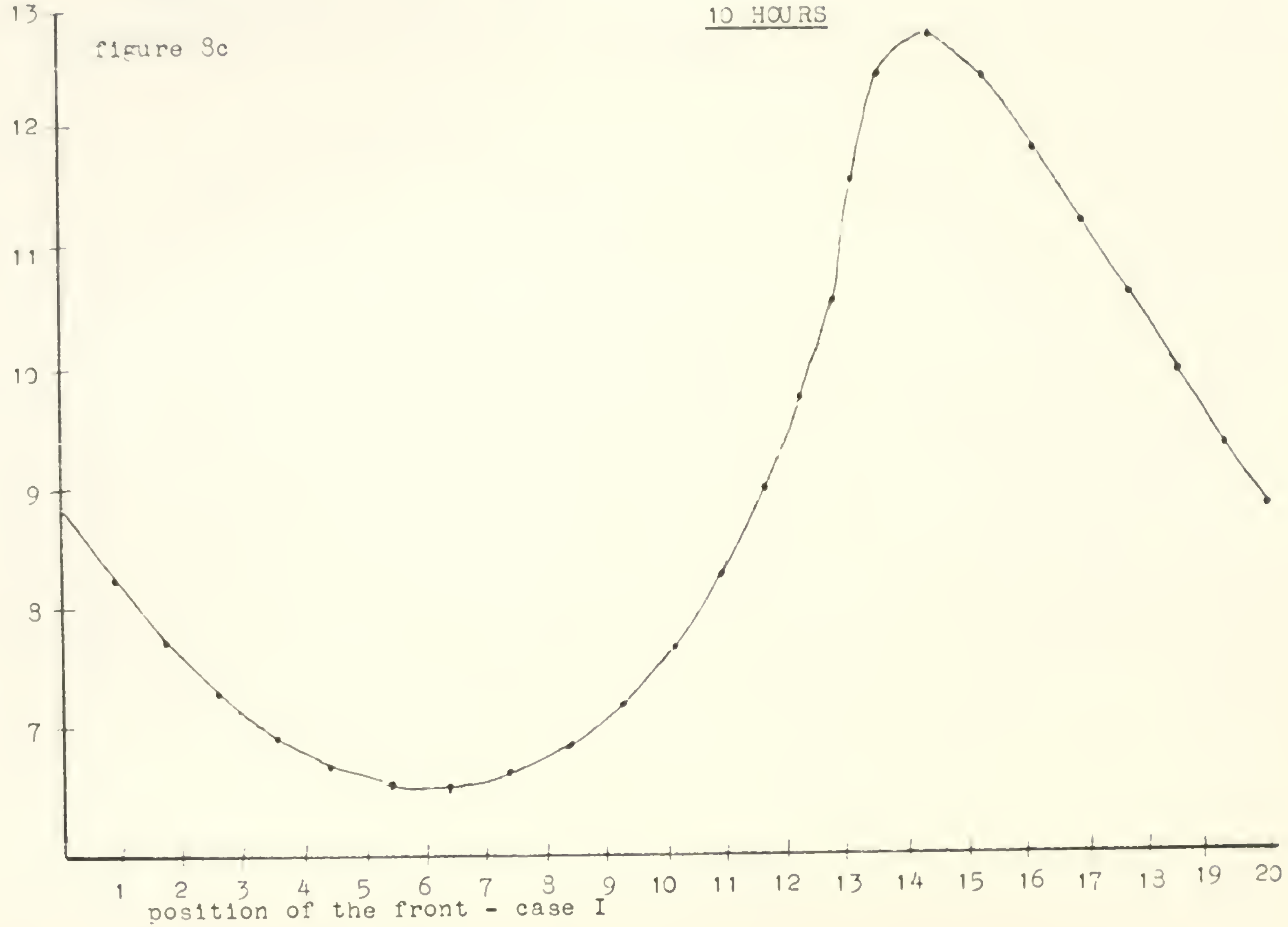
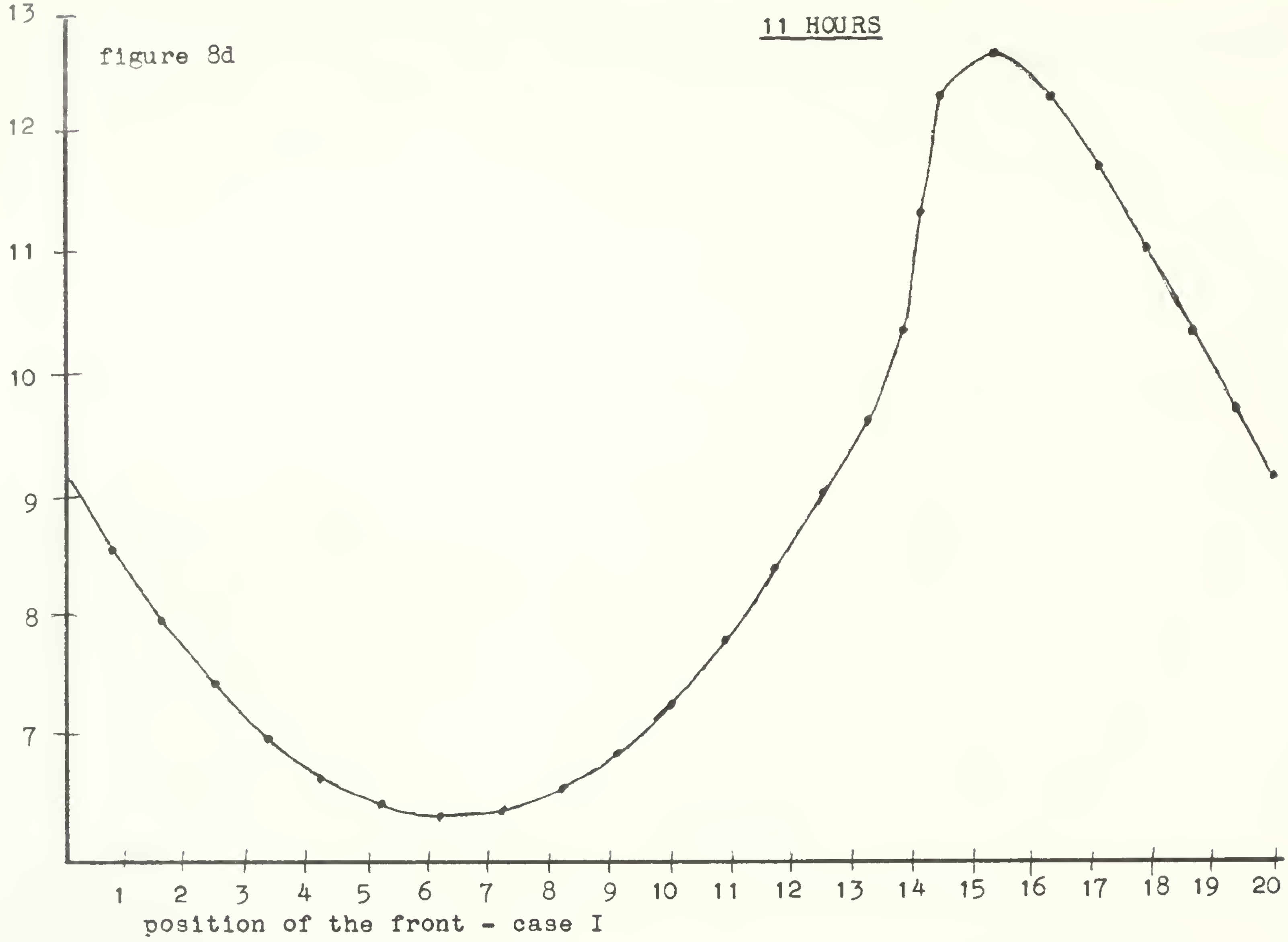

9 HOURS
figure 8b
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I

figure 8c



10 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I



11 HOURS
figure 8d
13
12
11
10
9
8
7
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
position of the front - case I

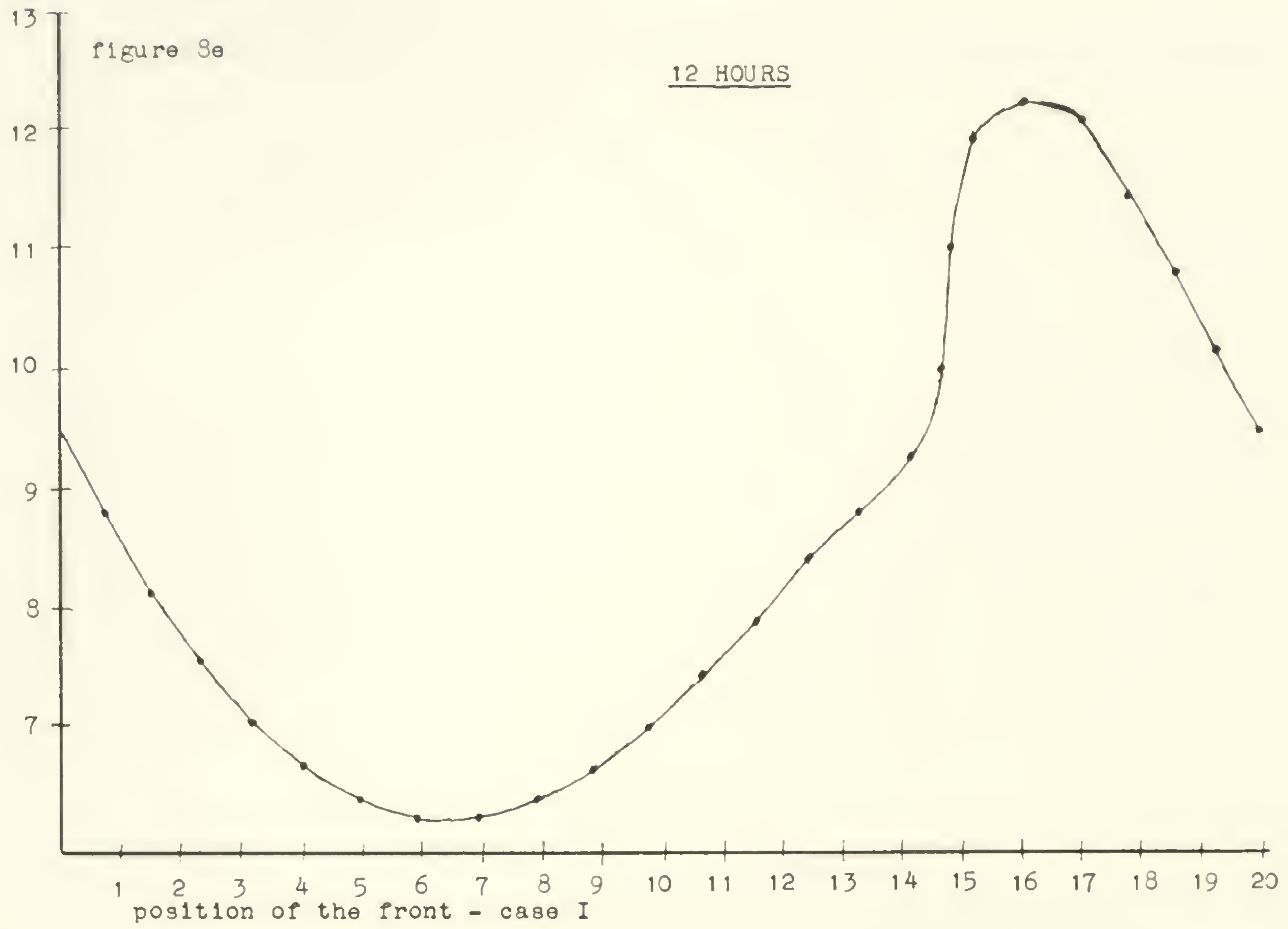


figure 8e
12 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I

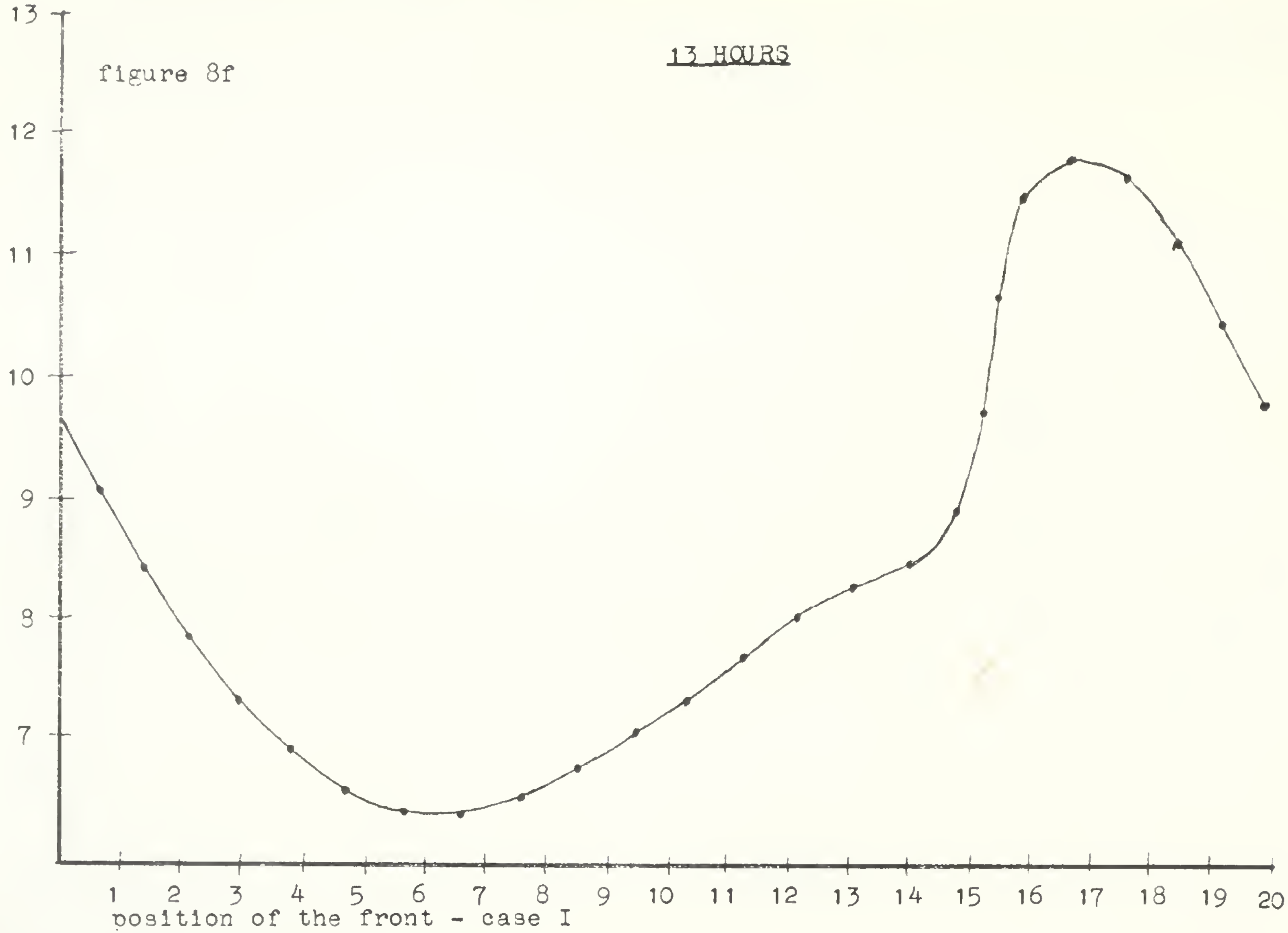

13
12
11
10
9
8
7
figure 8f
13 HOURS
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I

figure 8g

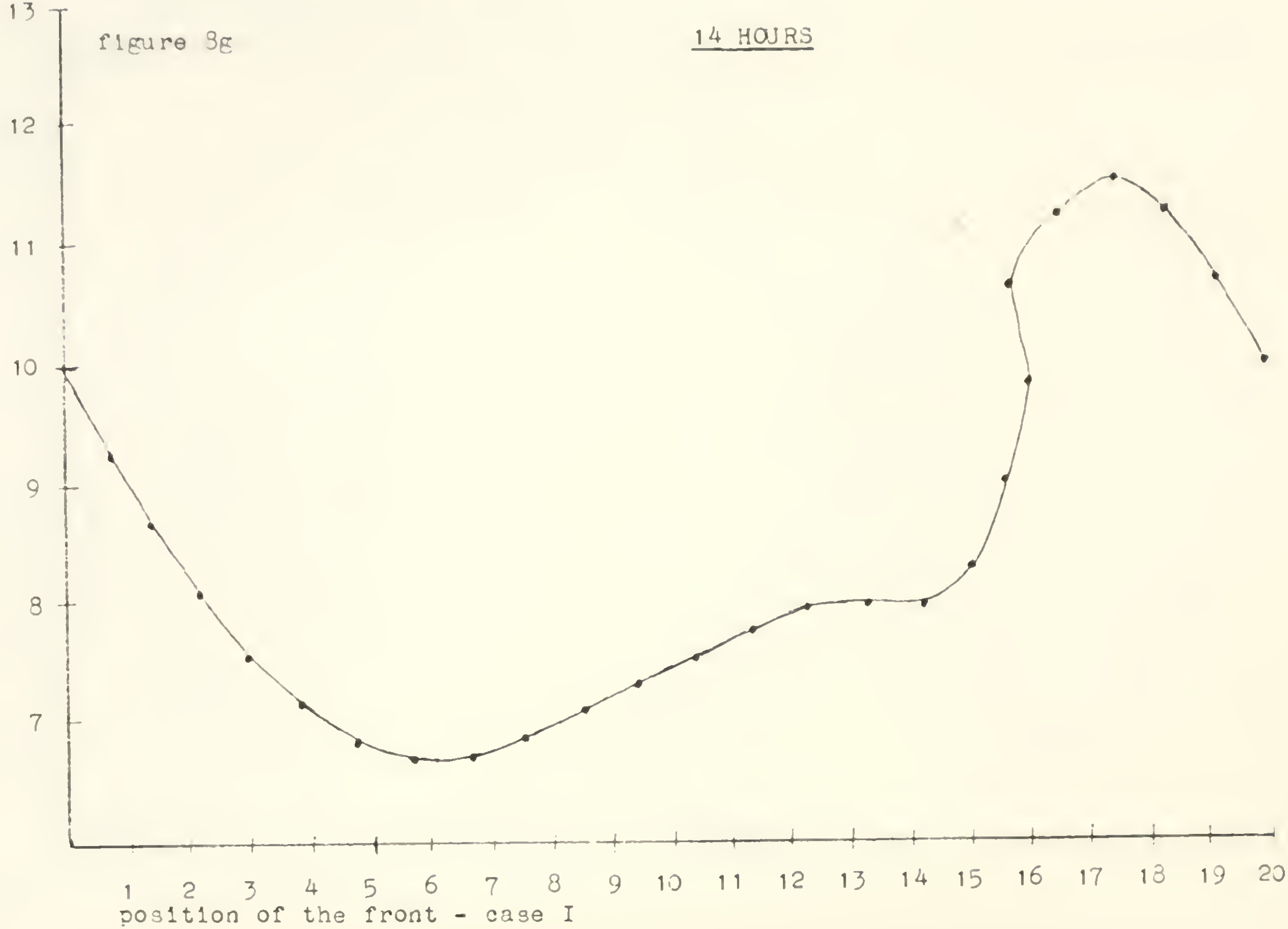
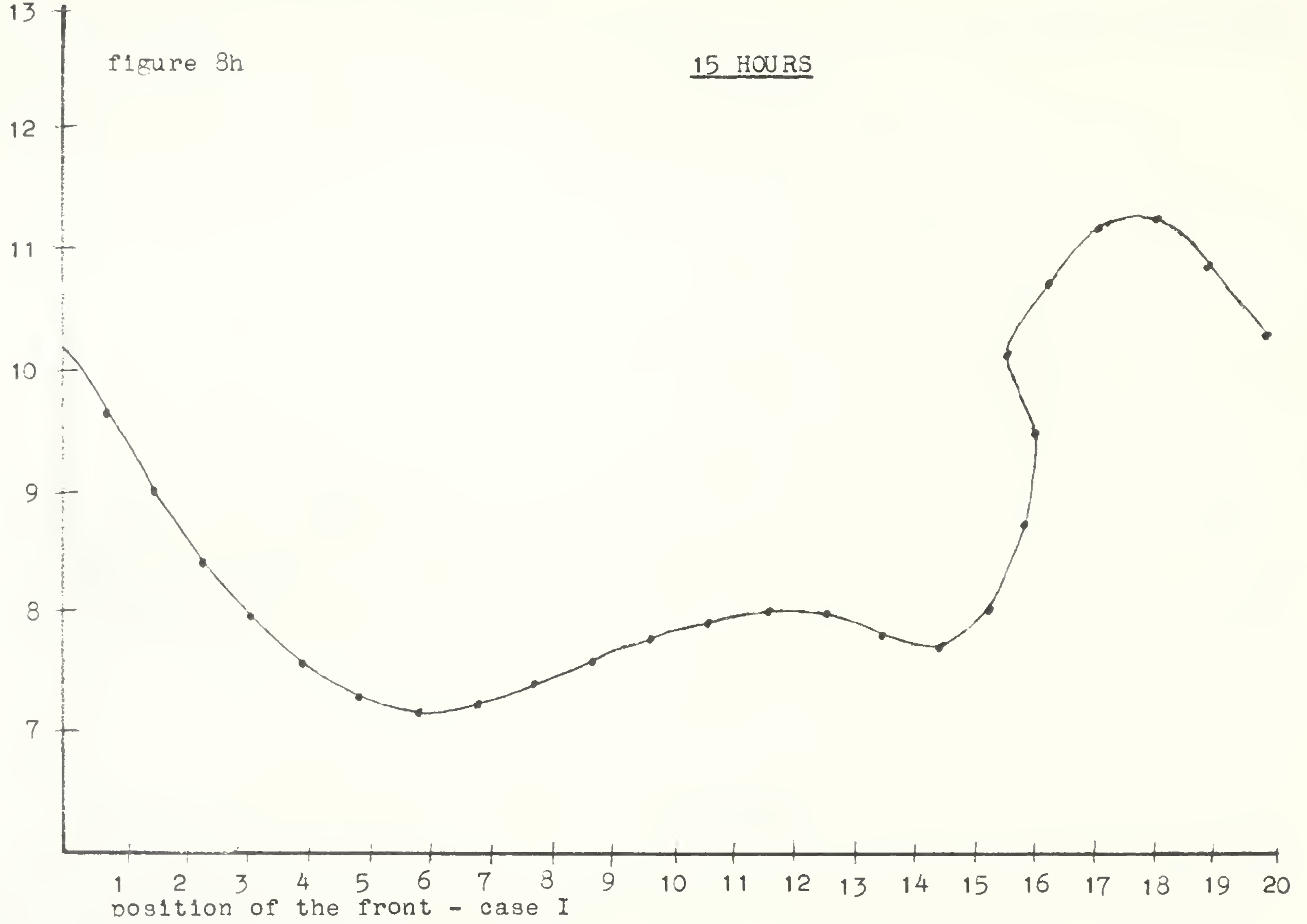

14 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I

figure 8h


15 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case I

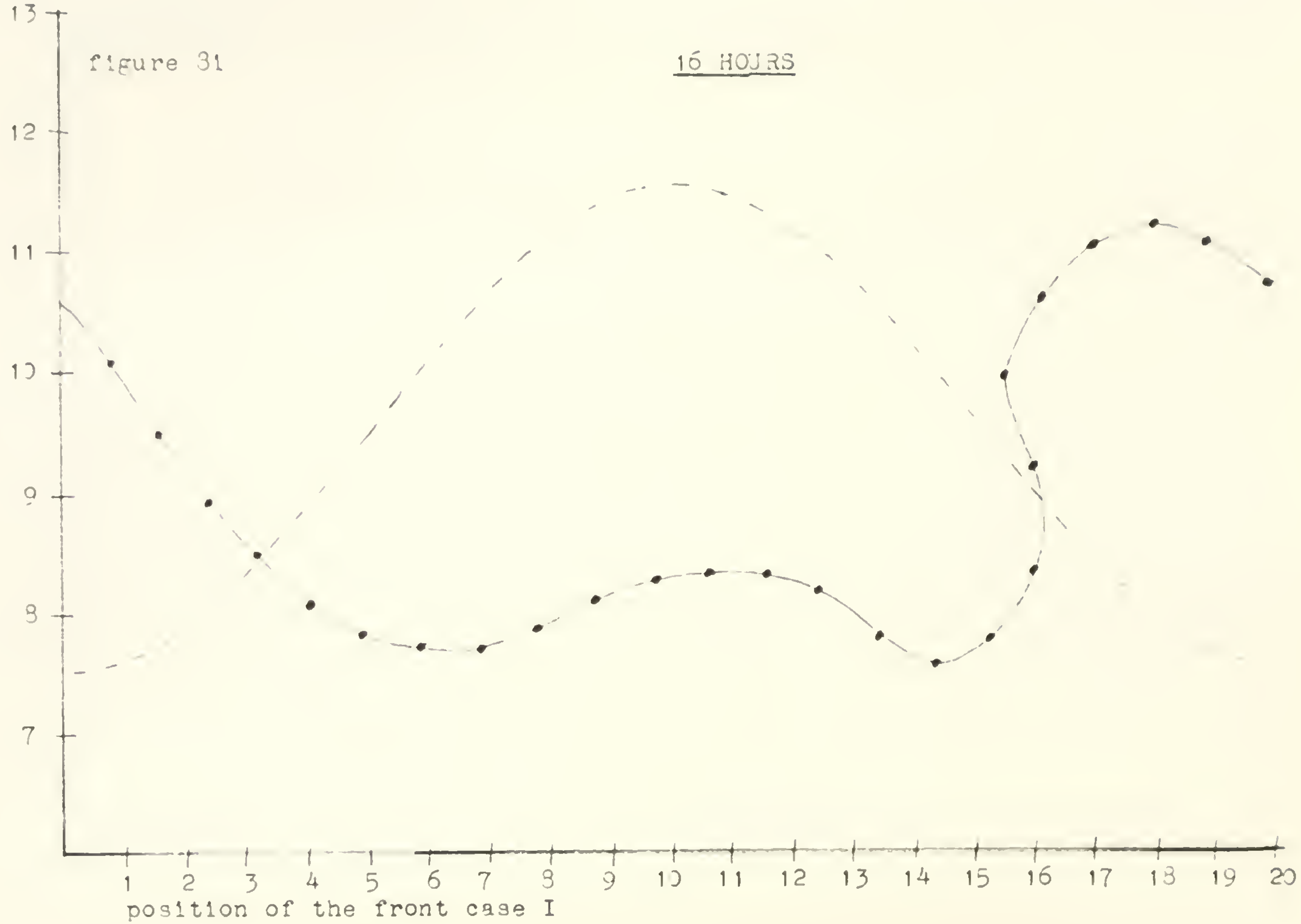
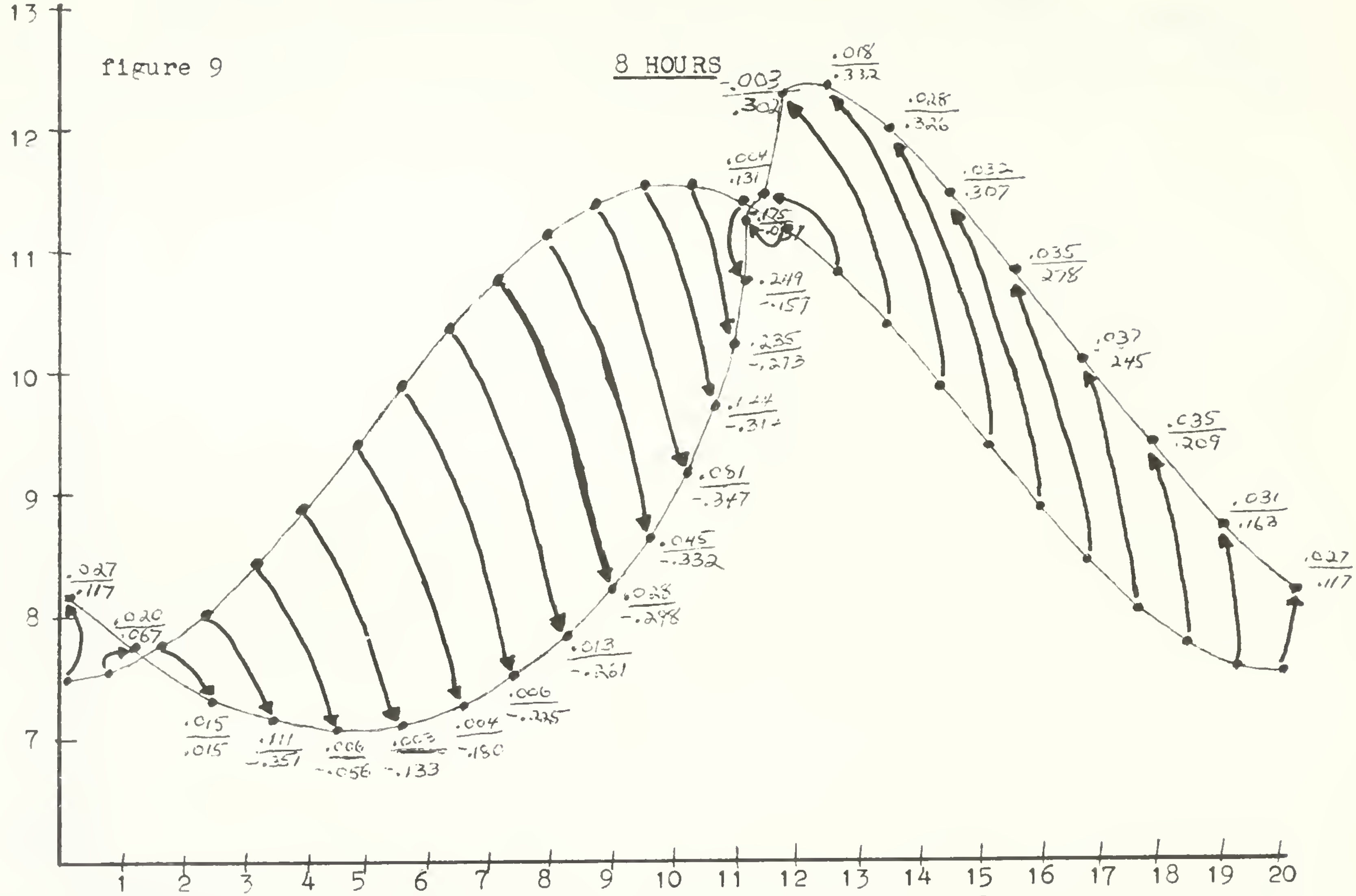
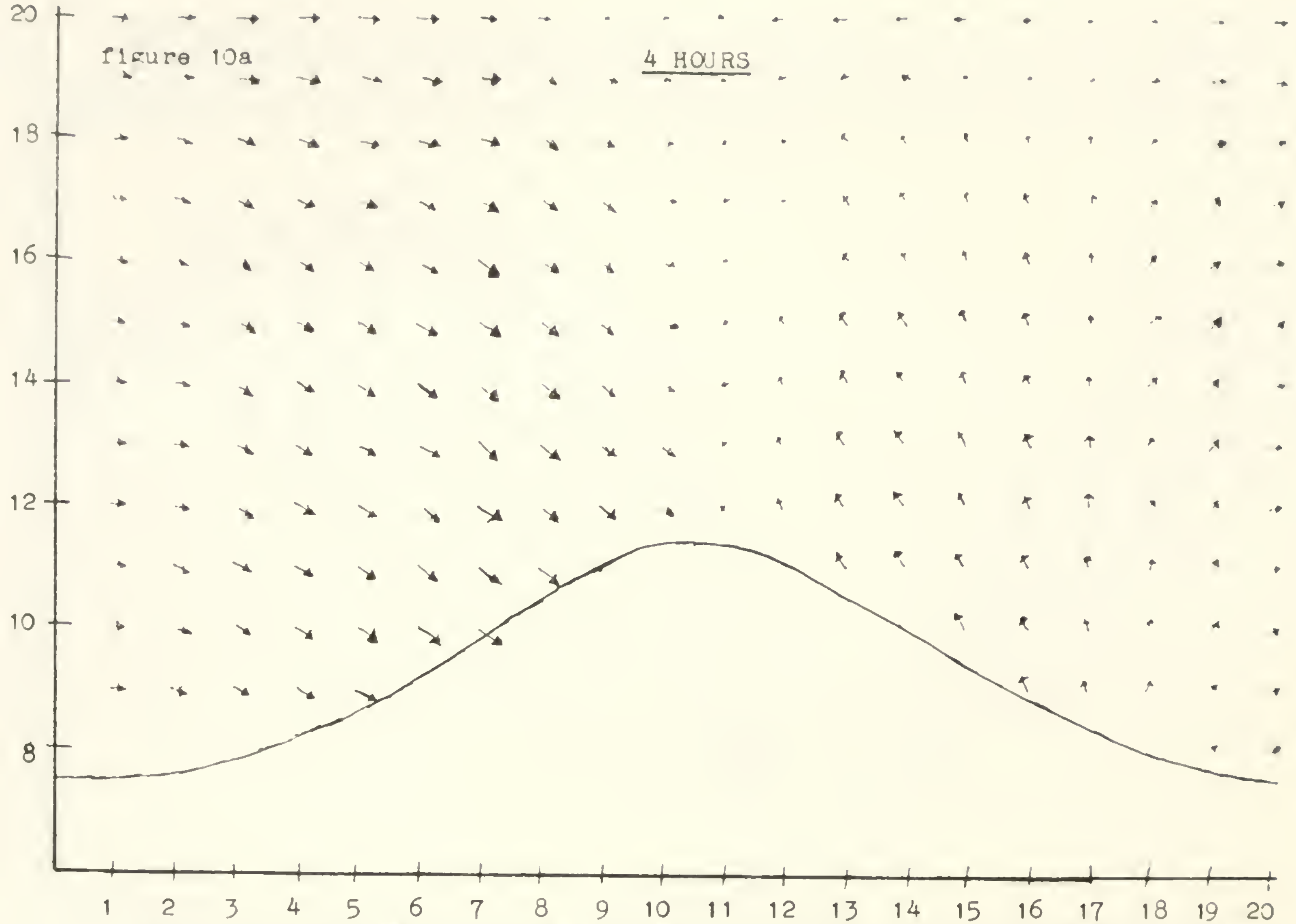


figure 31
16 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front case I

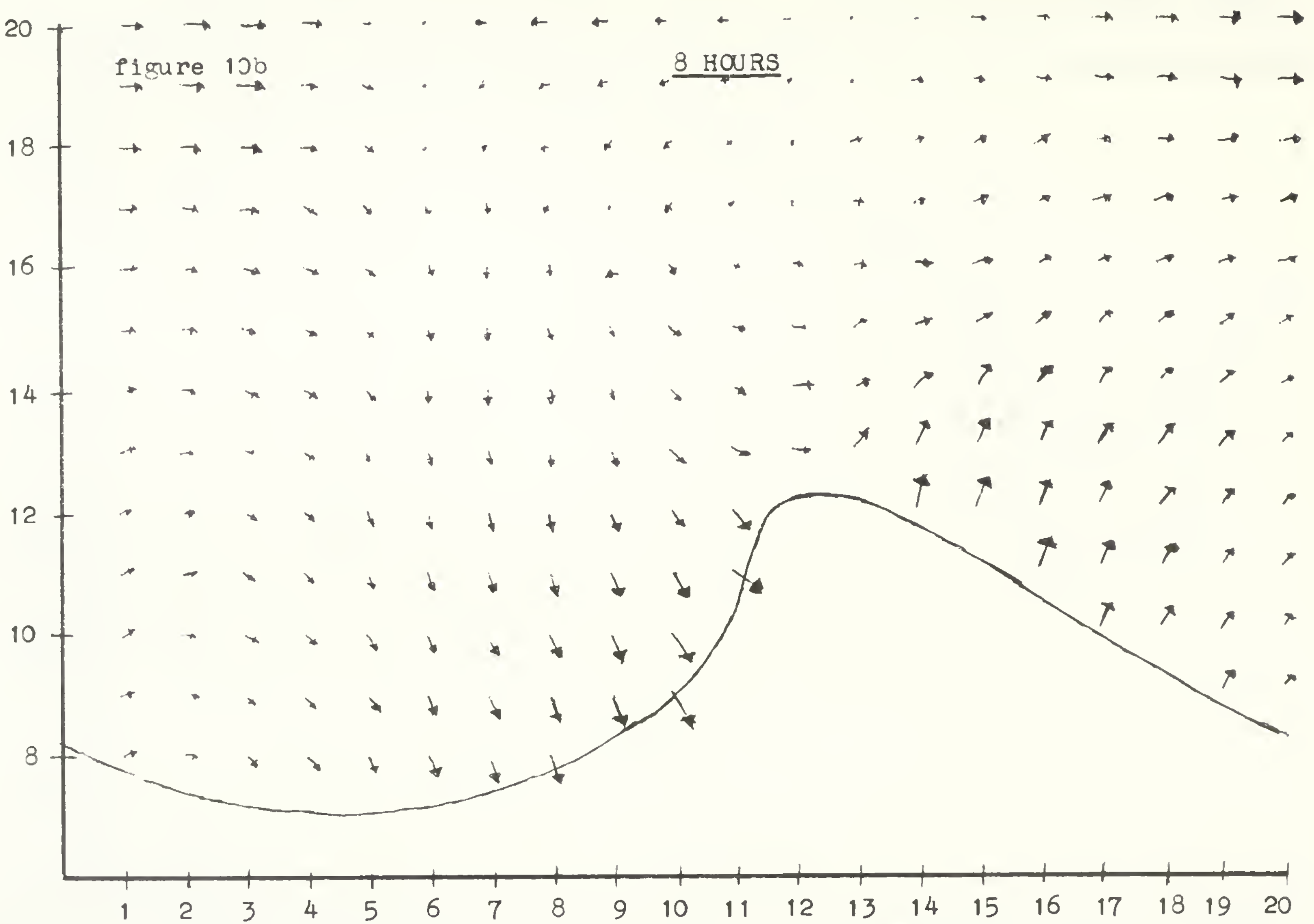
figure 9



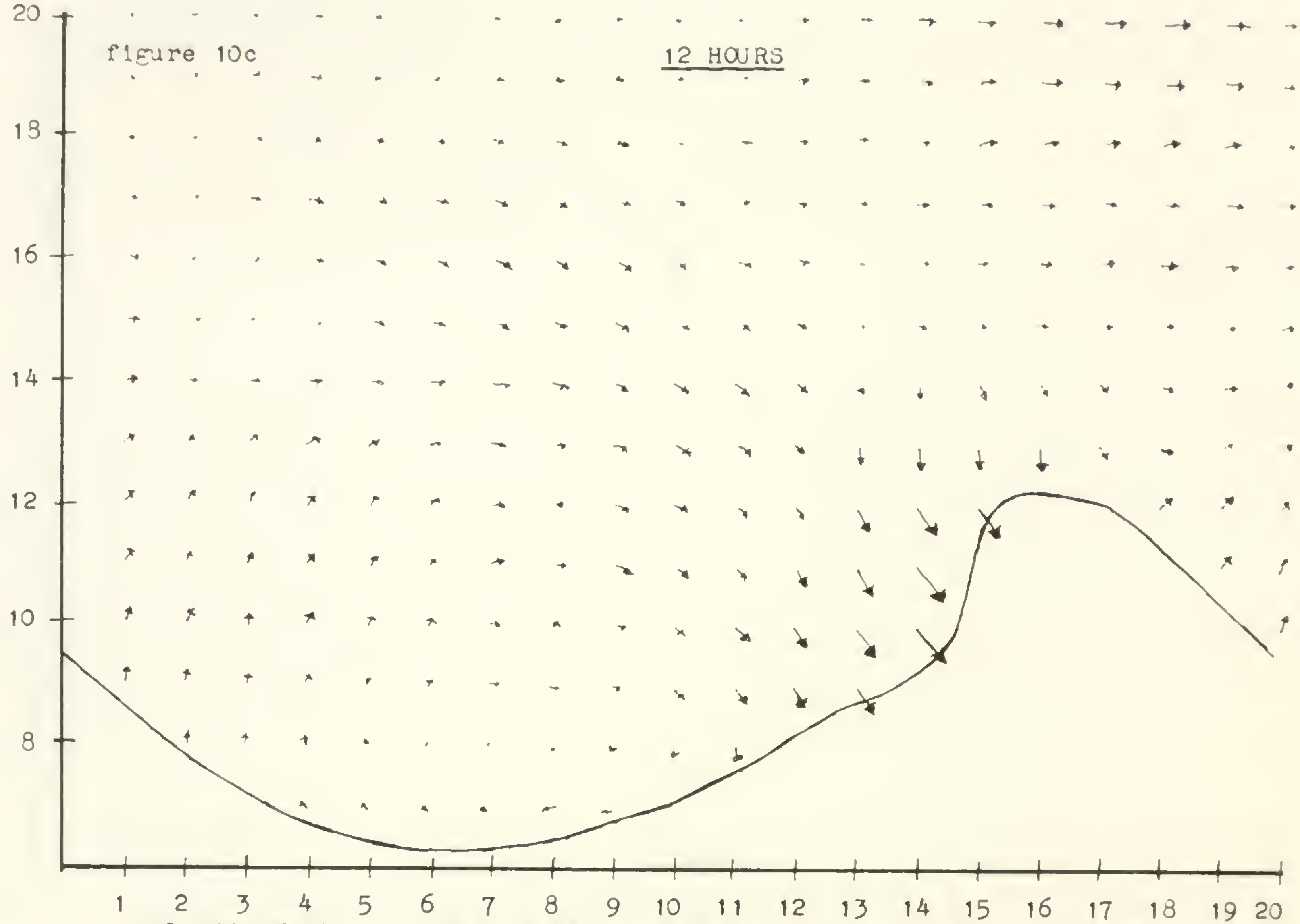
Trajectories of the front points during the period of 8 hours for case I.
No redistribution of the front points was made during this 8 hour period.



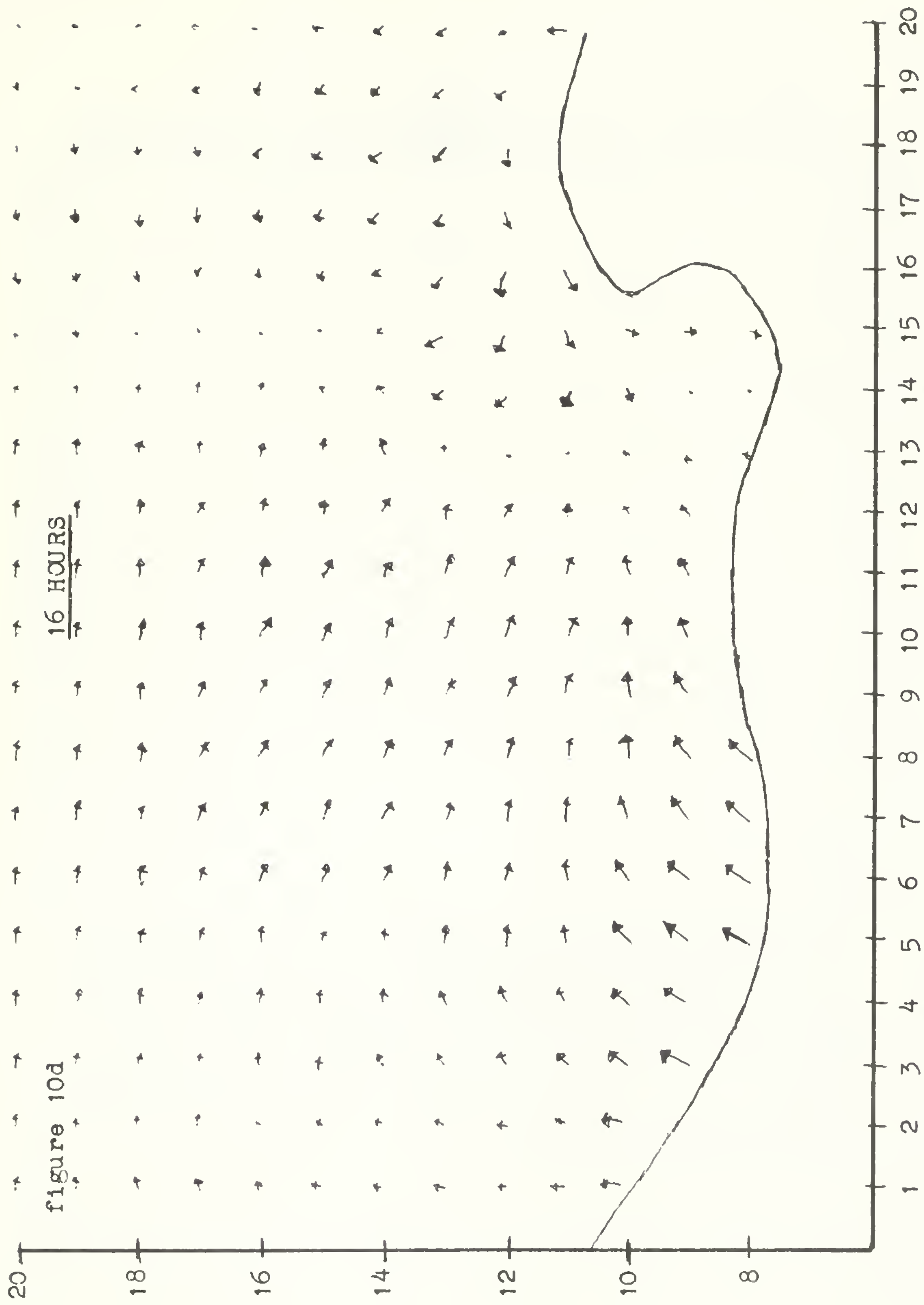
Vector velocity field, in terms of the original fixed coordinate system, for case I. The tail of the arrow is at the net point and the length of the arrow is proportional to the magnitude of the velocity.



velocity field in the cold air - case I in the fixed coordinate system

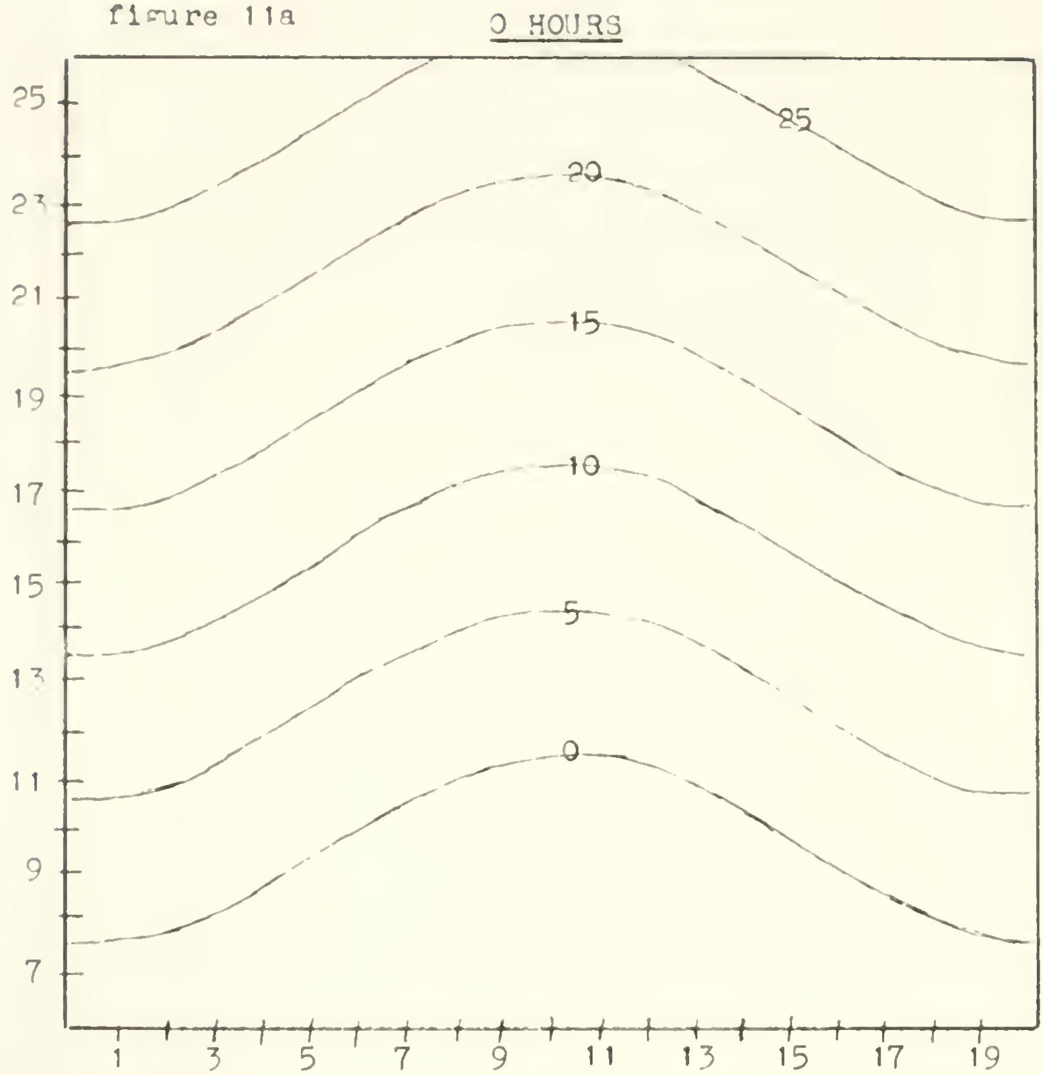


velocity field in the cold air - case I in the fixed coordinate system



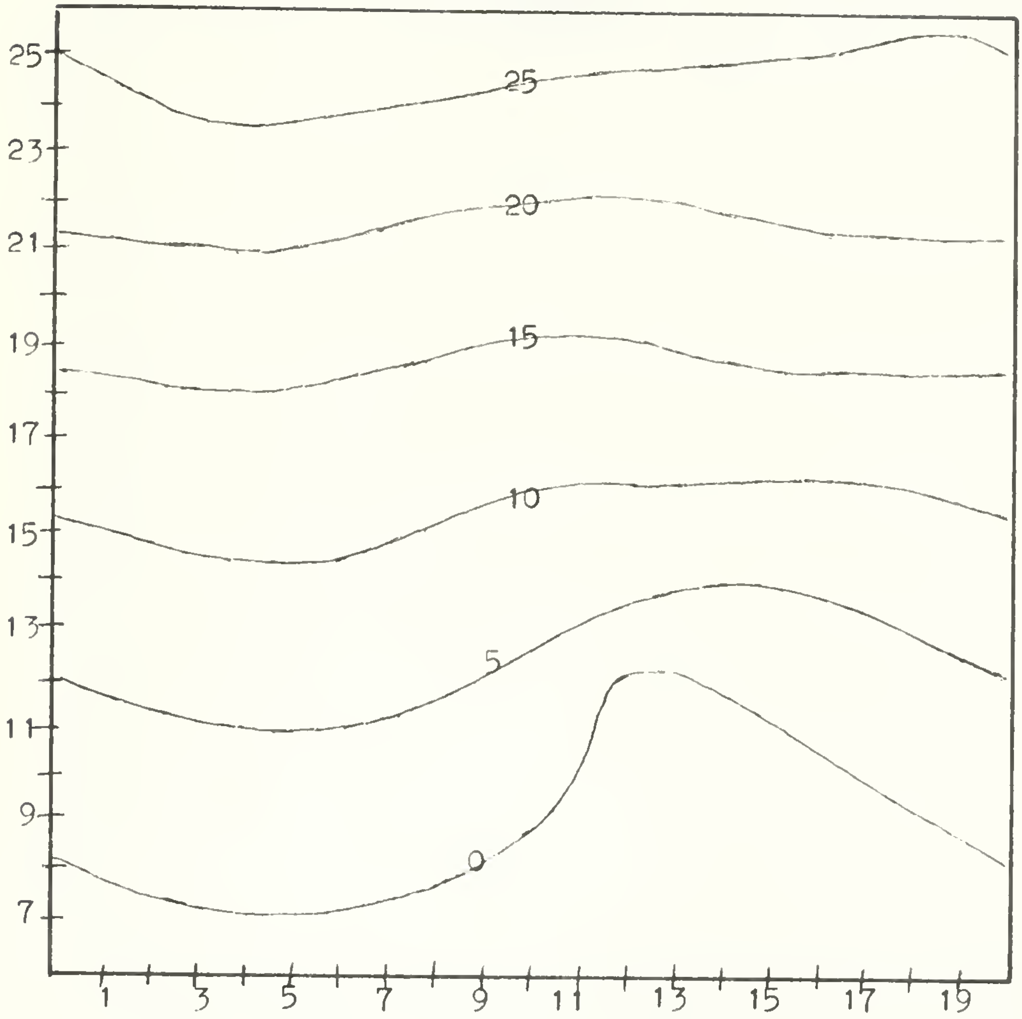
velocity field in the cold air - case I in the fixed coordinate system

figure 11a



Initial height contour pattern of the cold air for case I. The contour lines are drawn at 5,000 foot intervals. Y is equal to 26Δs so as to correspond to the graphs in K.I.S.

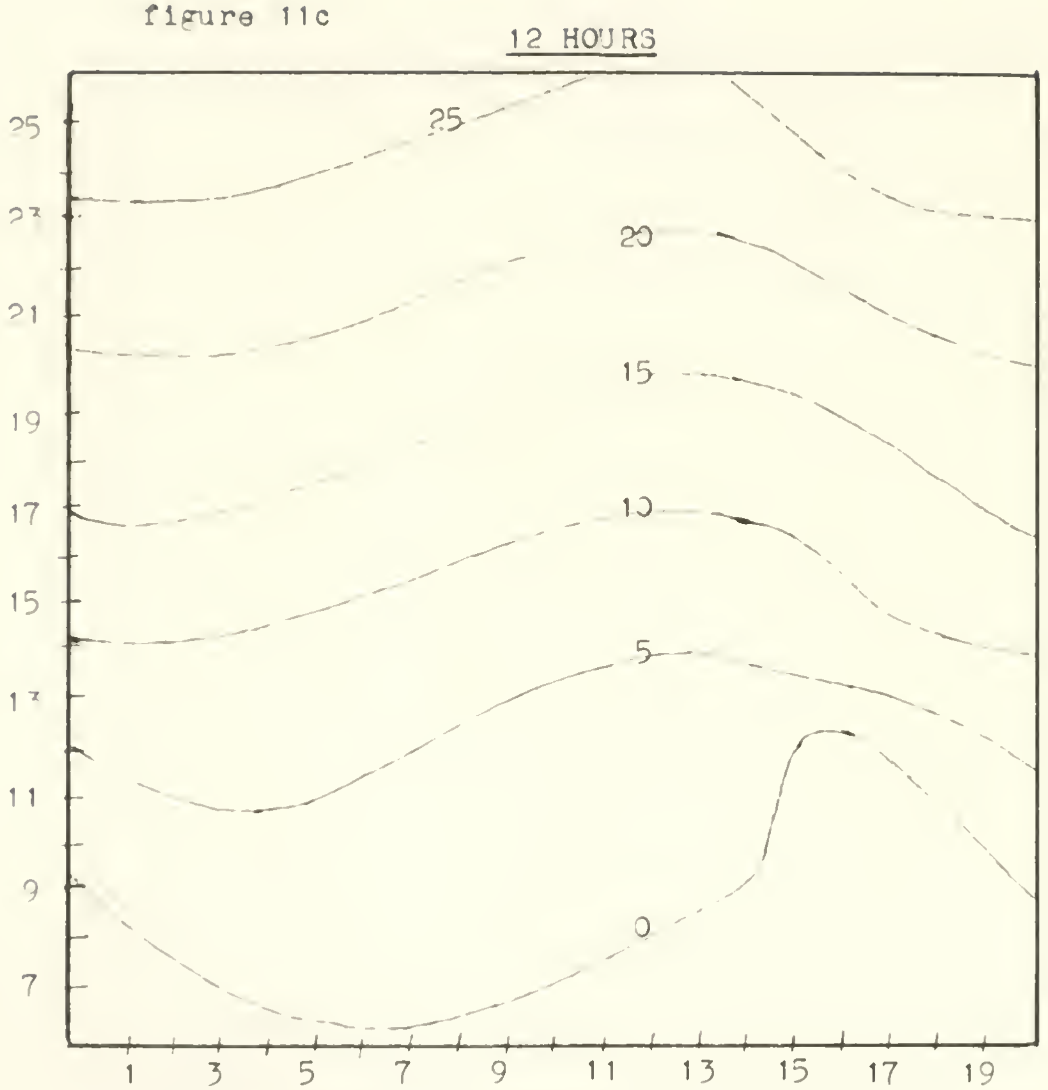
figure 11b 8 HOURS



Height contour pattern of the cold air for case I.

The contour lines are drawn at 5,000 foot intervals.

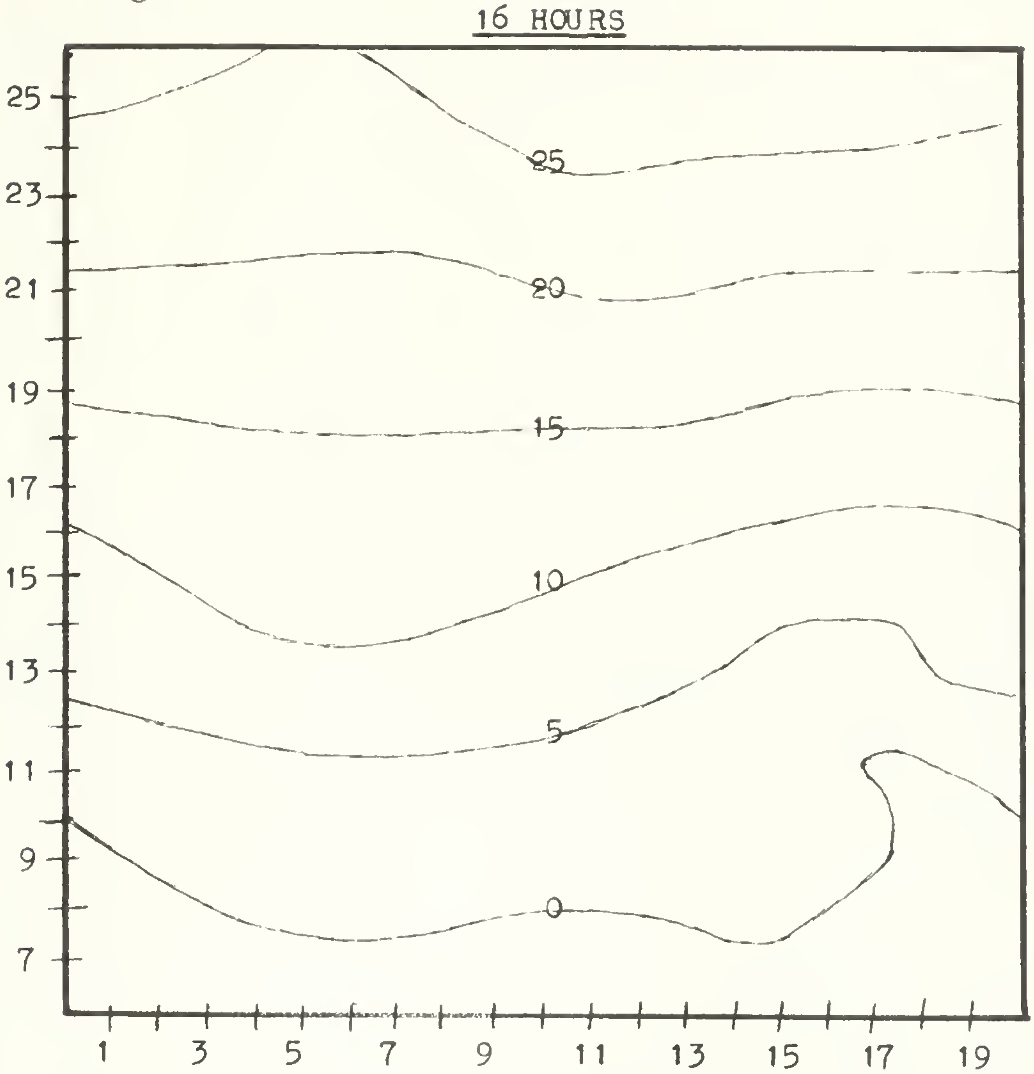
figure 11c



Height contor pattern of the cold air for case I.

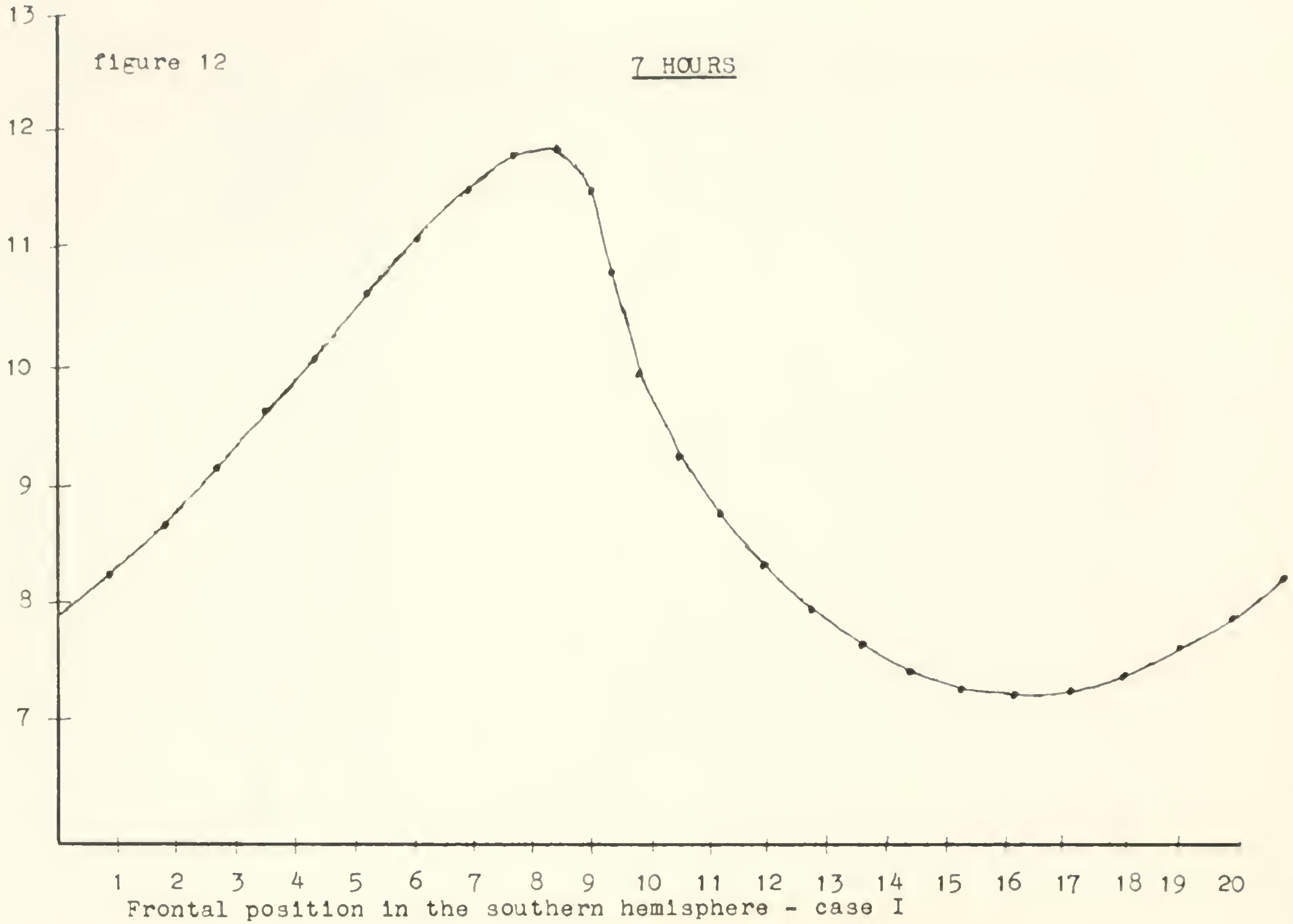
The contour lines are drawn at 5,000 foot intervals.

figure 11d



Height contour pattern of the cold air for case I. The contour lines are drawn at 5,000 foot intervals.

figure 12


7 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Frontal position in the southern hemisphere - case I

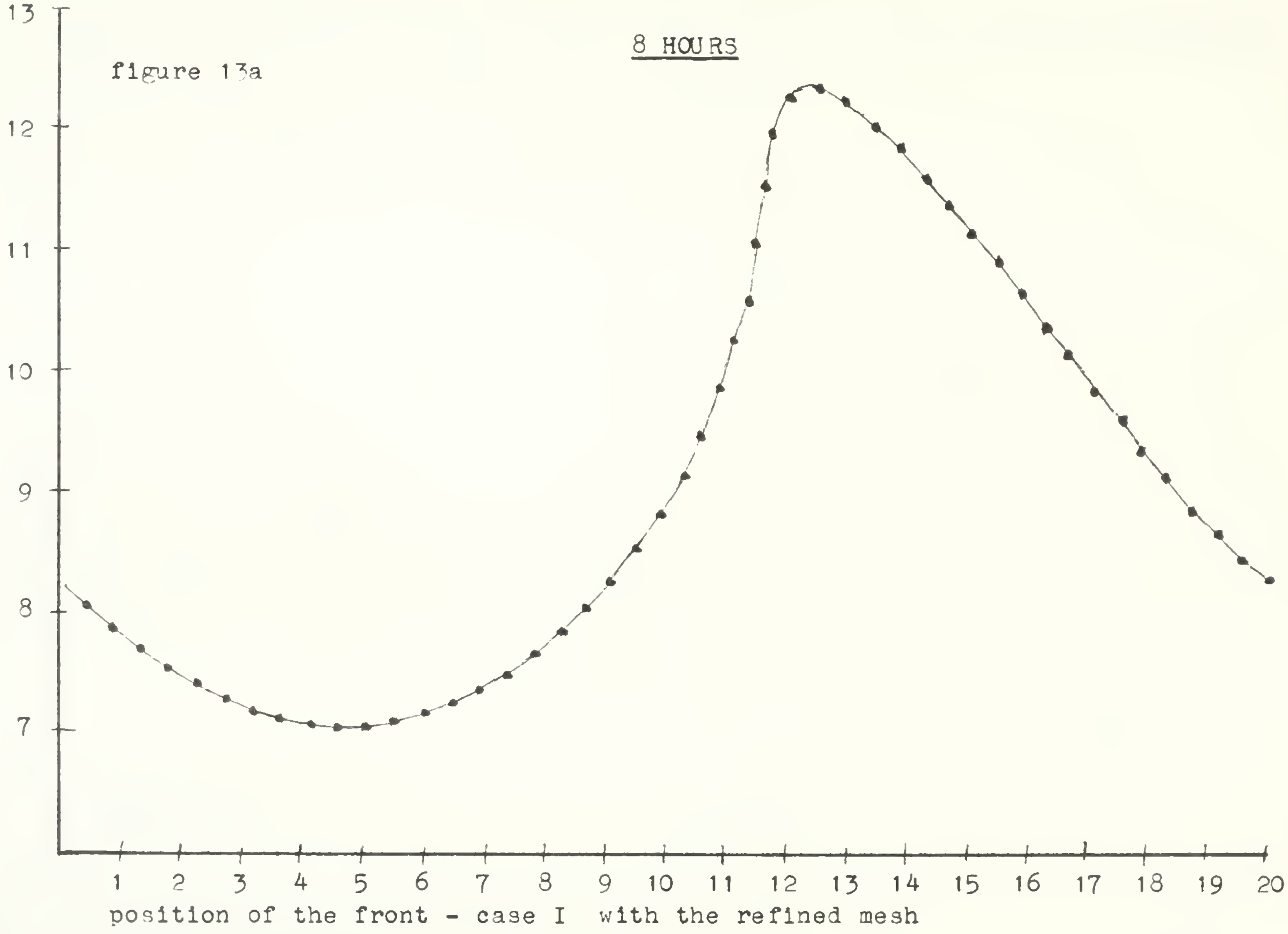
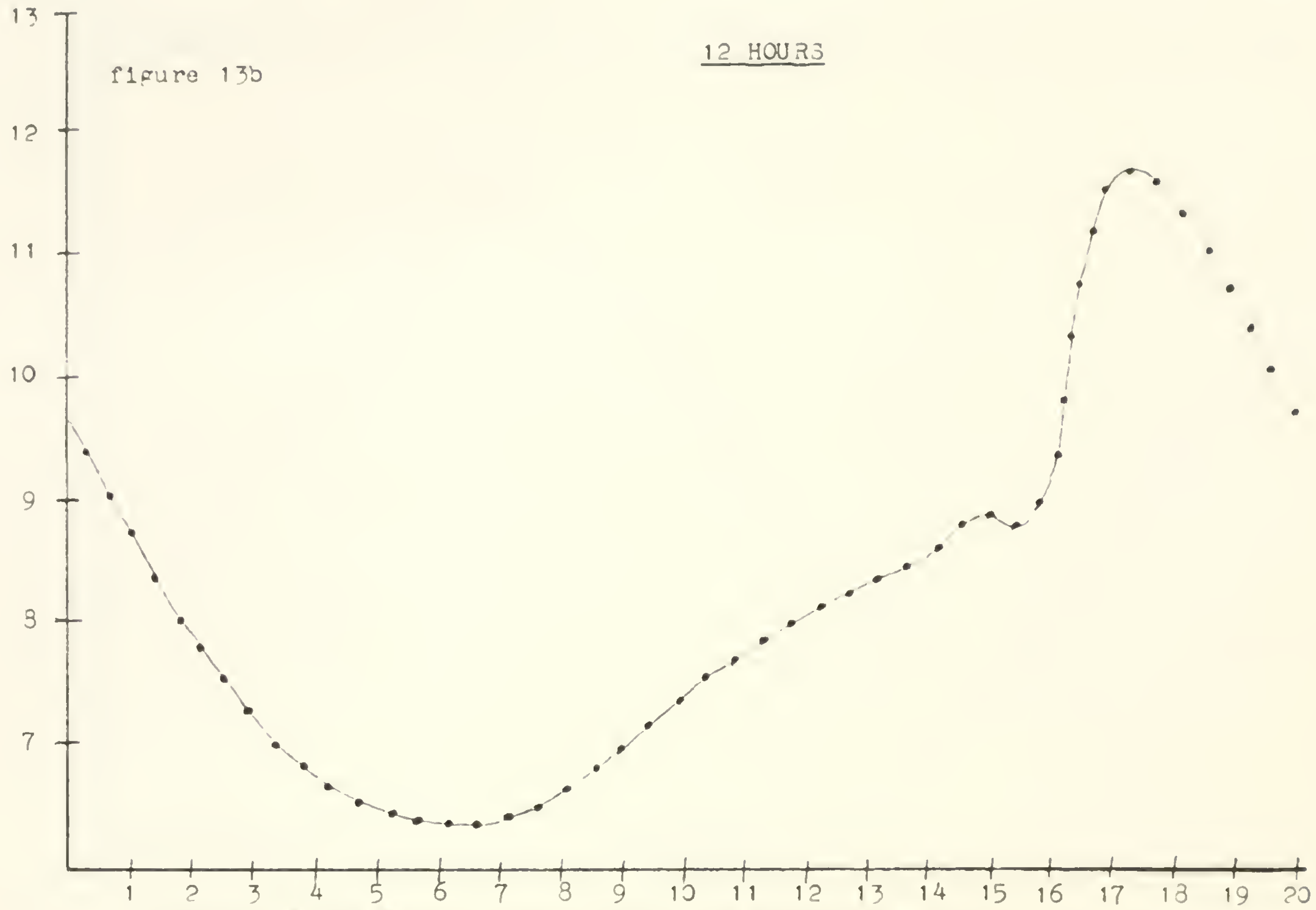


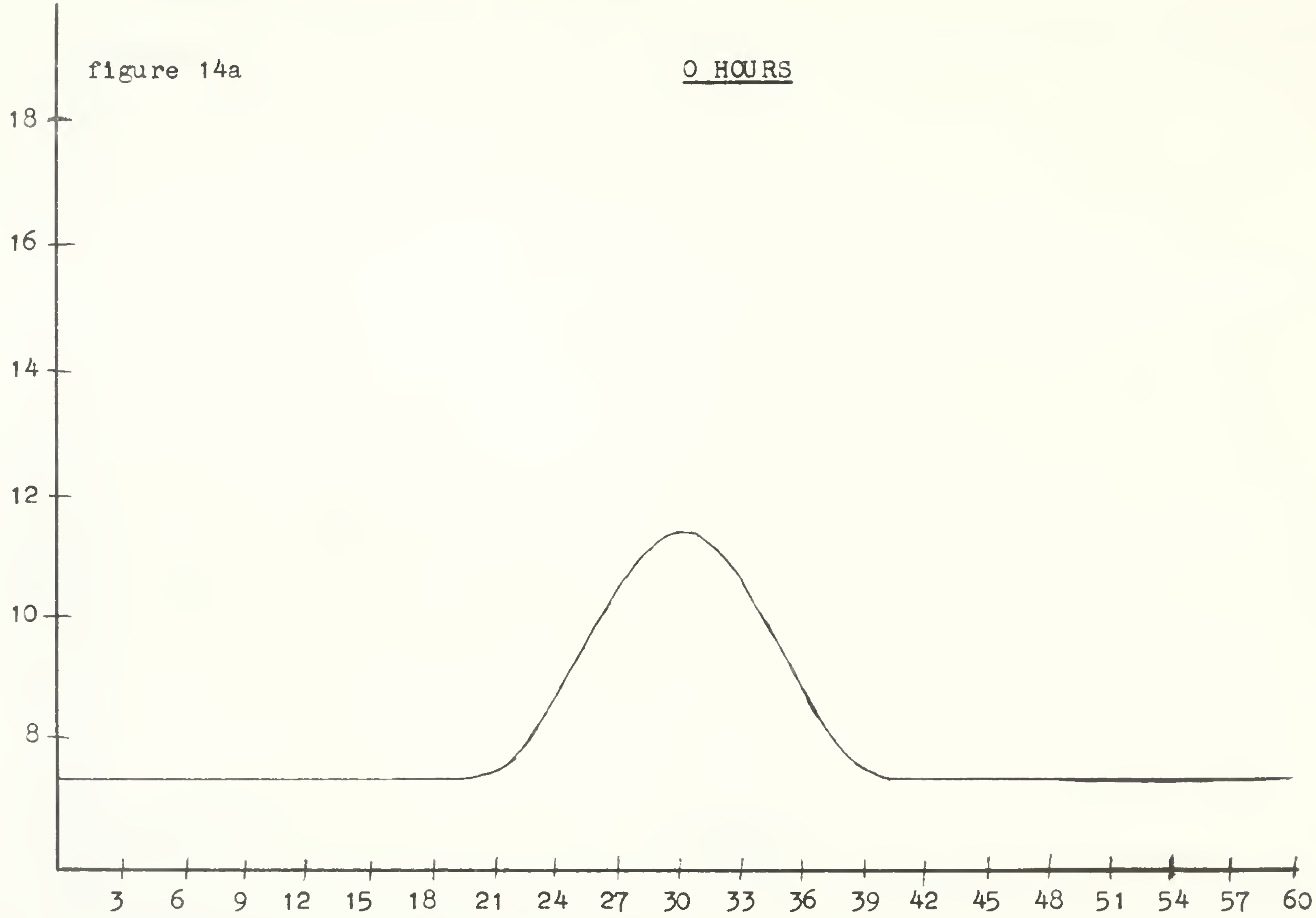
figure 13a
8 HOURS
13
12
11
10
9
8
7
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
position of the front - case I with the refined mesh

figure 13b



position of the front - case I with the refined mesh.

figure 14a



Initial position of the front for case III.

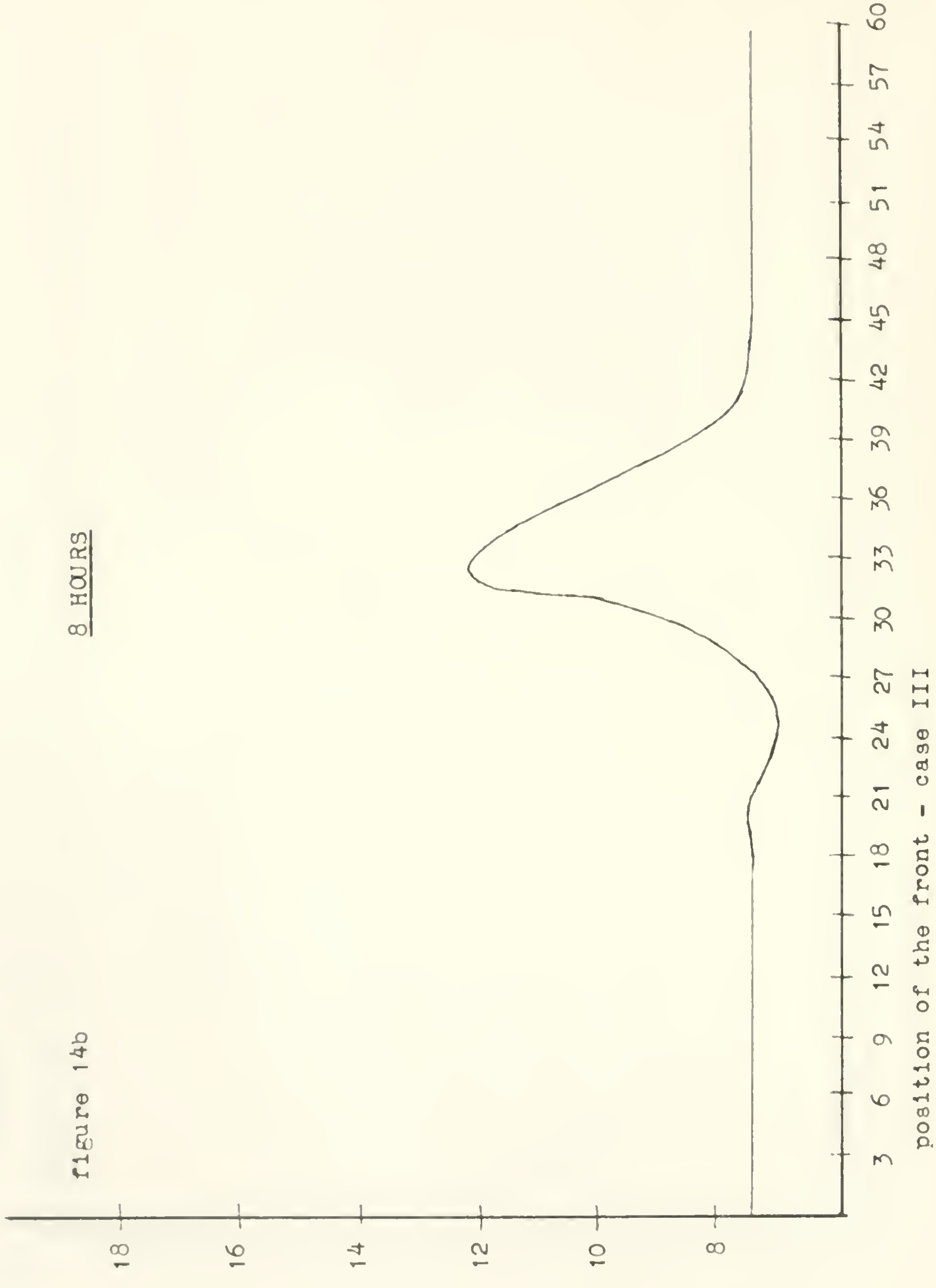

8 HOURS
18
16
14
12
10
8
3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60
position of the front - case III
figure 14b

figure 14c

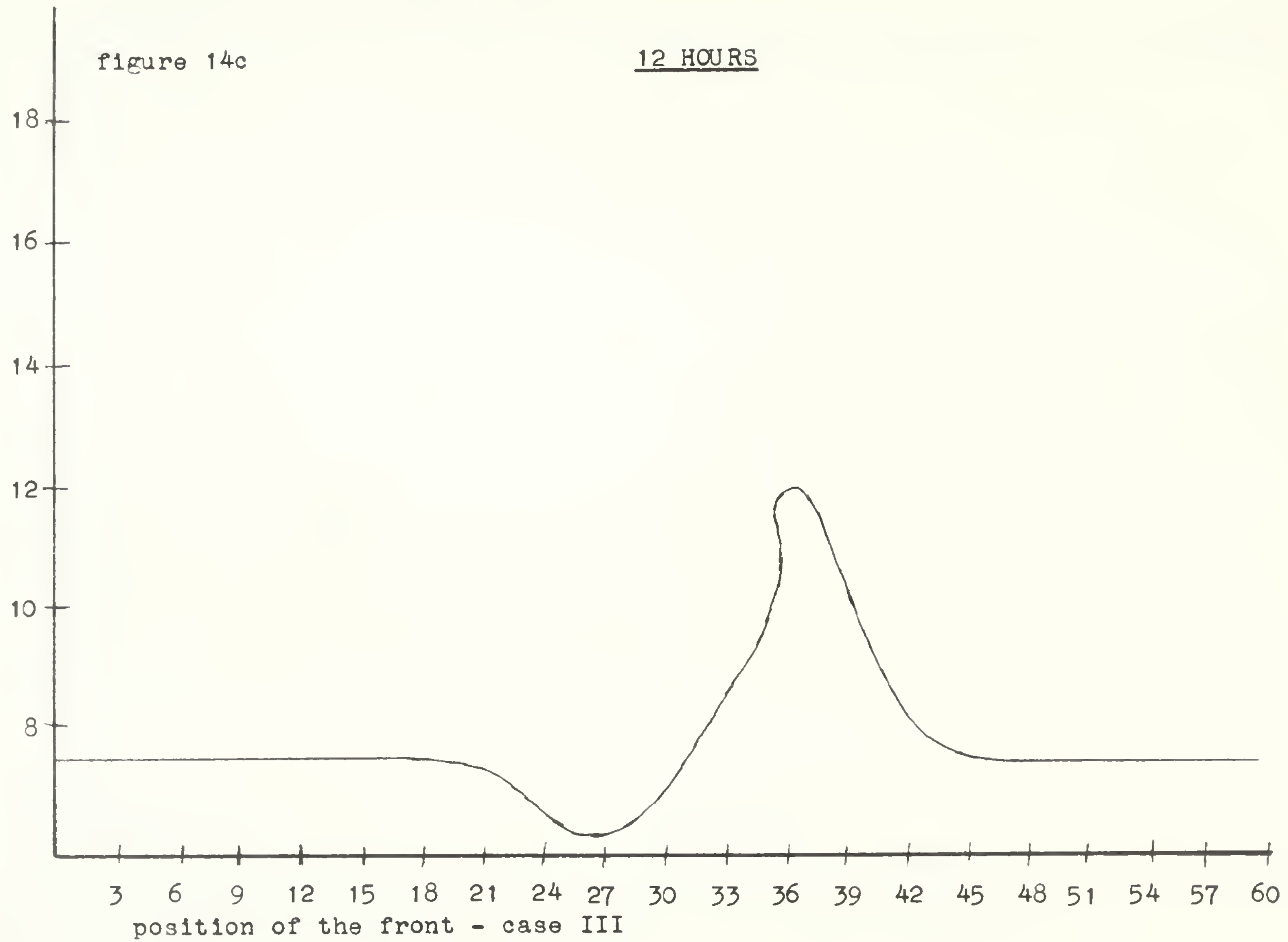
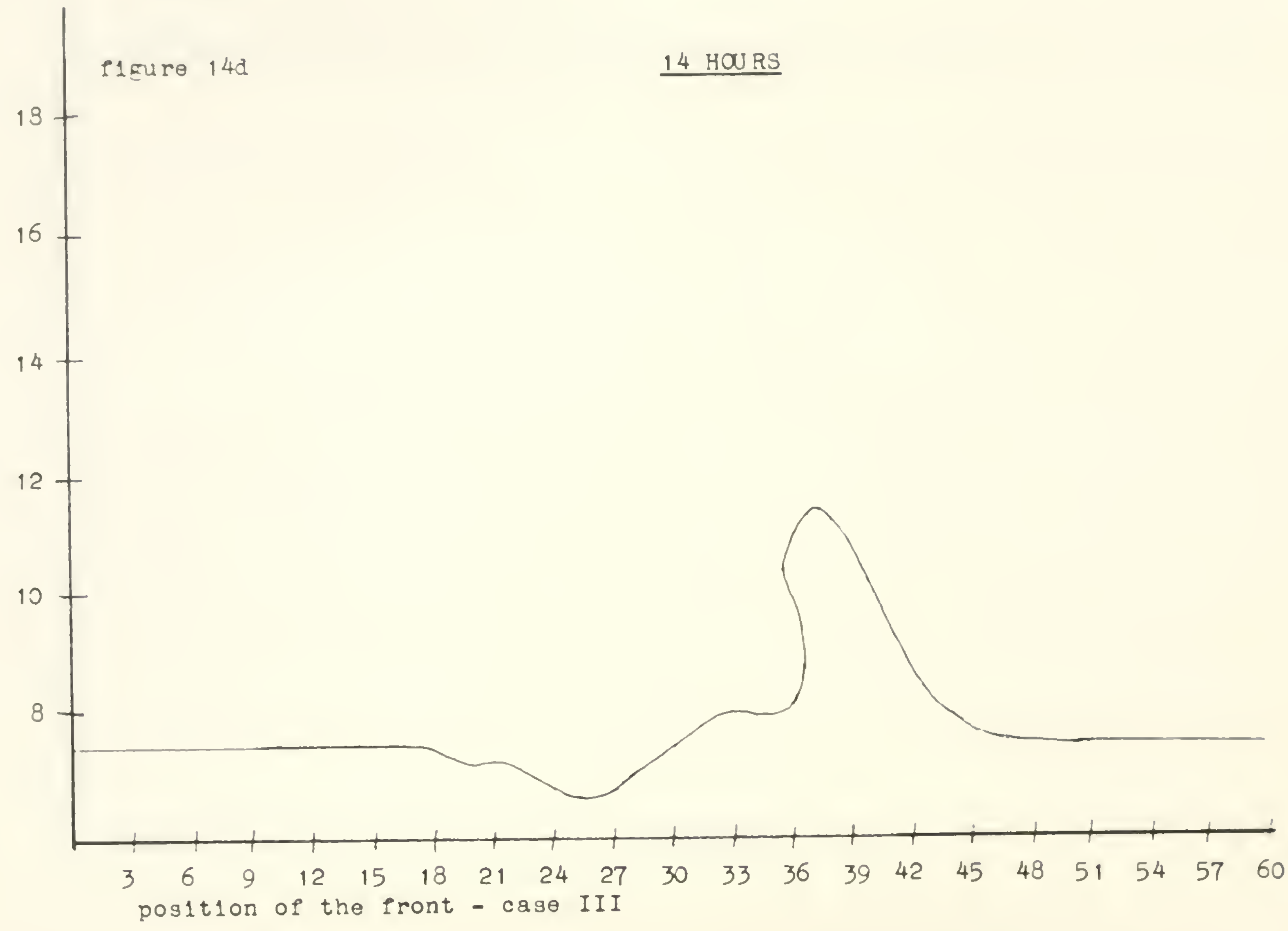

12 HOURS
18
16
14
12
10
8
3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60
position of the front - case III

figure 14d

14 HOURS



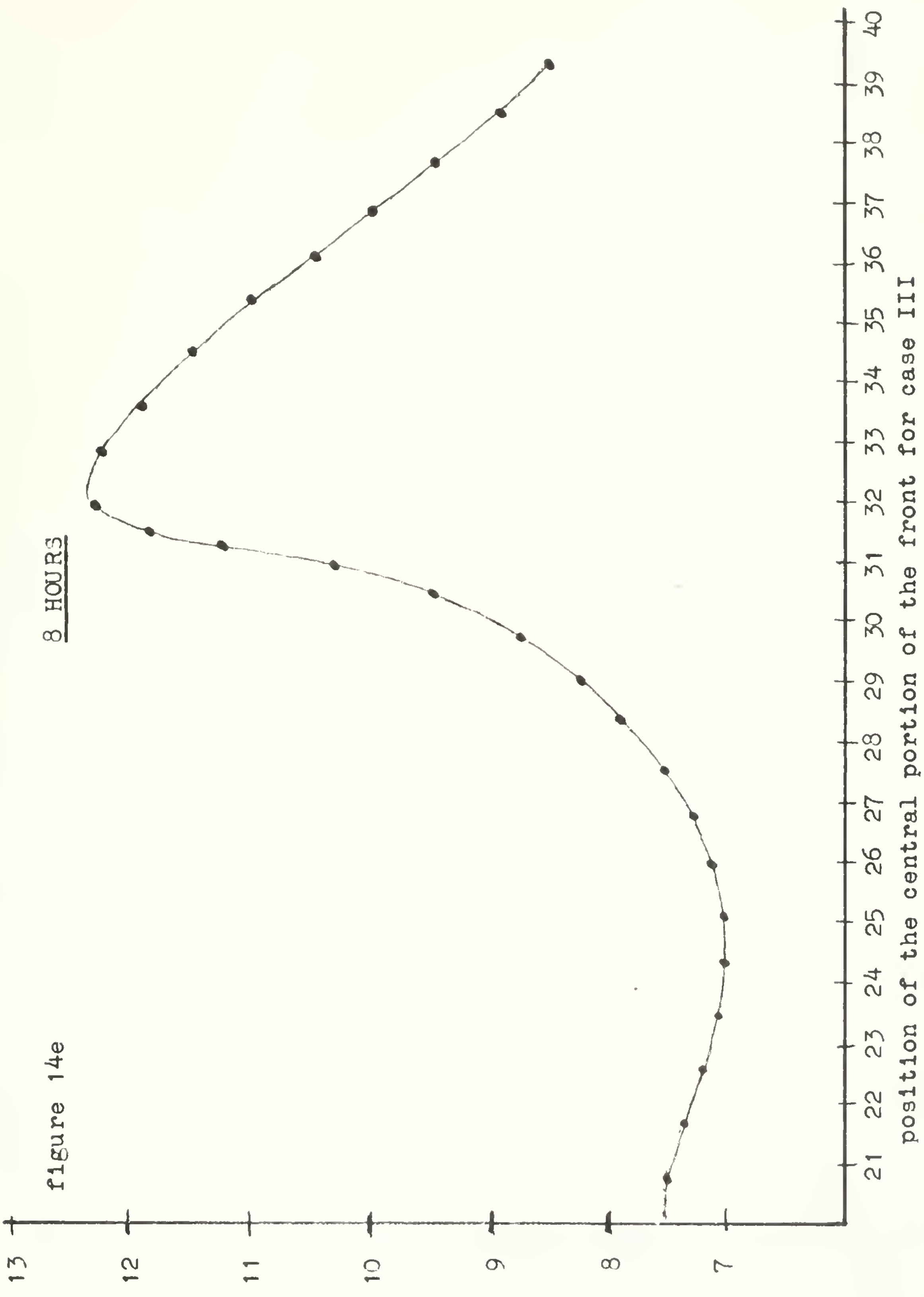


figure 14e
8 HOURS
13
12
11
10
9
8
7
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
position of the central portion of the front for case III

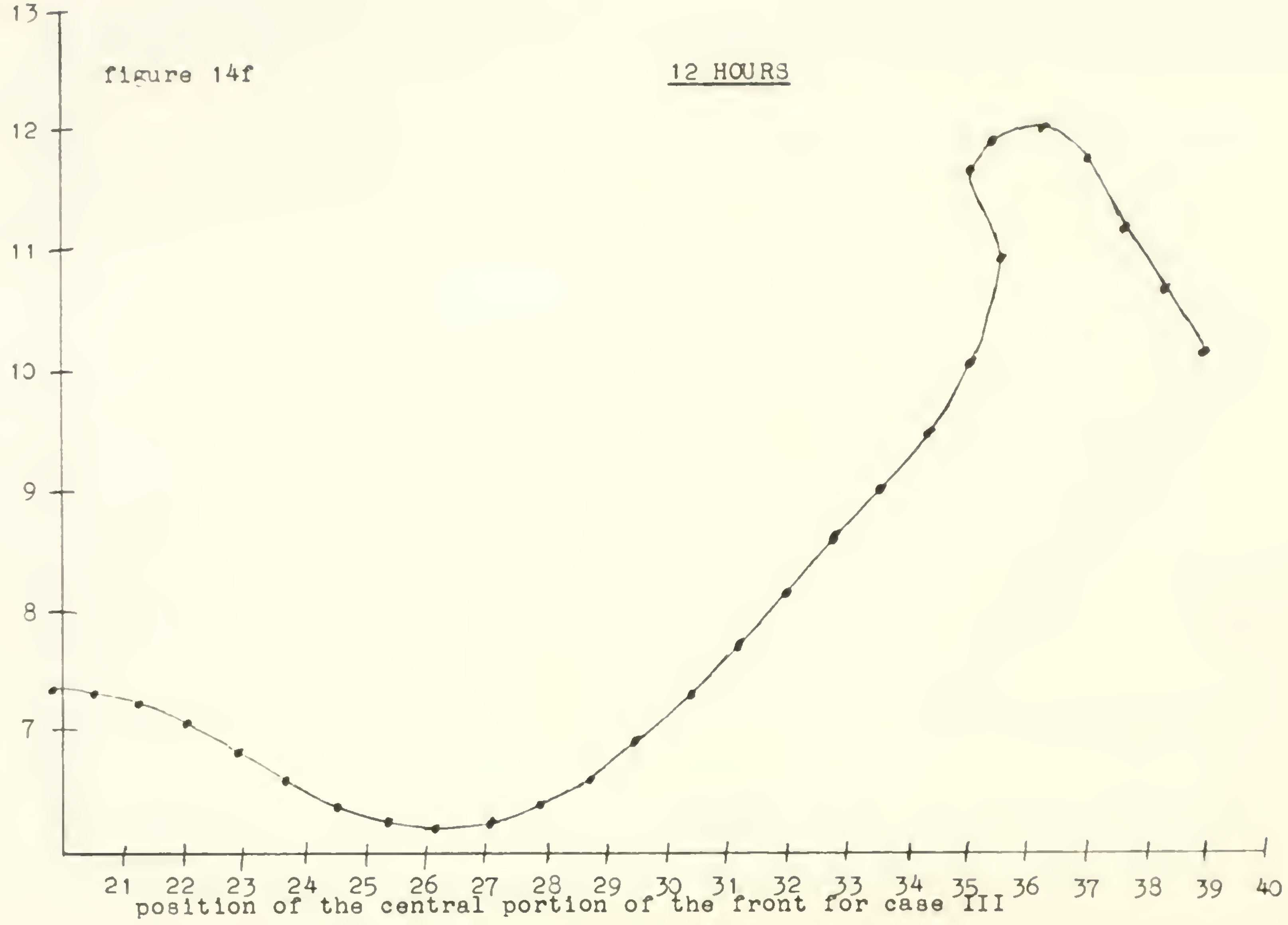

12 HOURS
13
12
11
10
9
8
7
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
position of the central portion of the front for case III

figure 14f

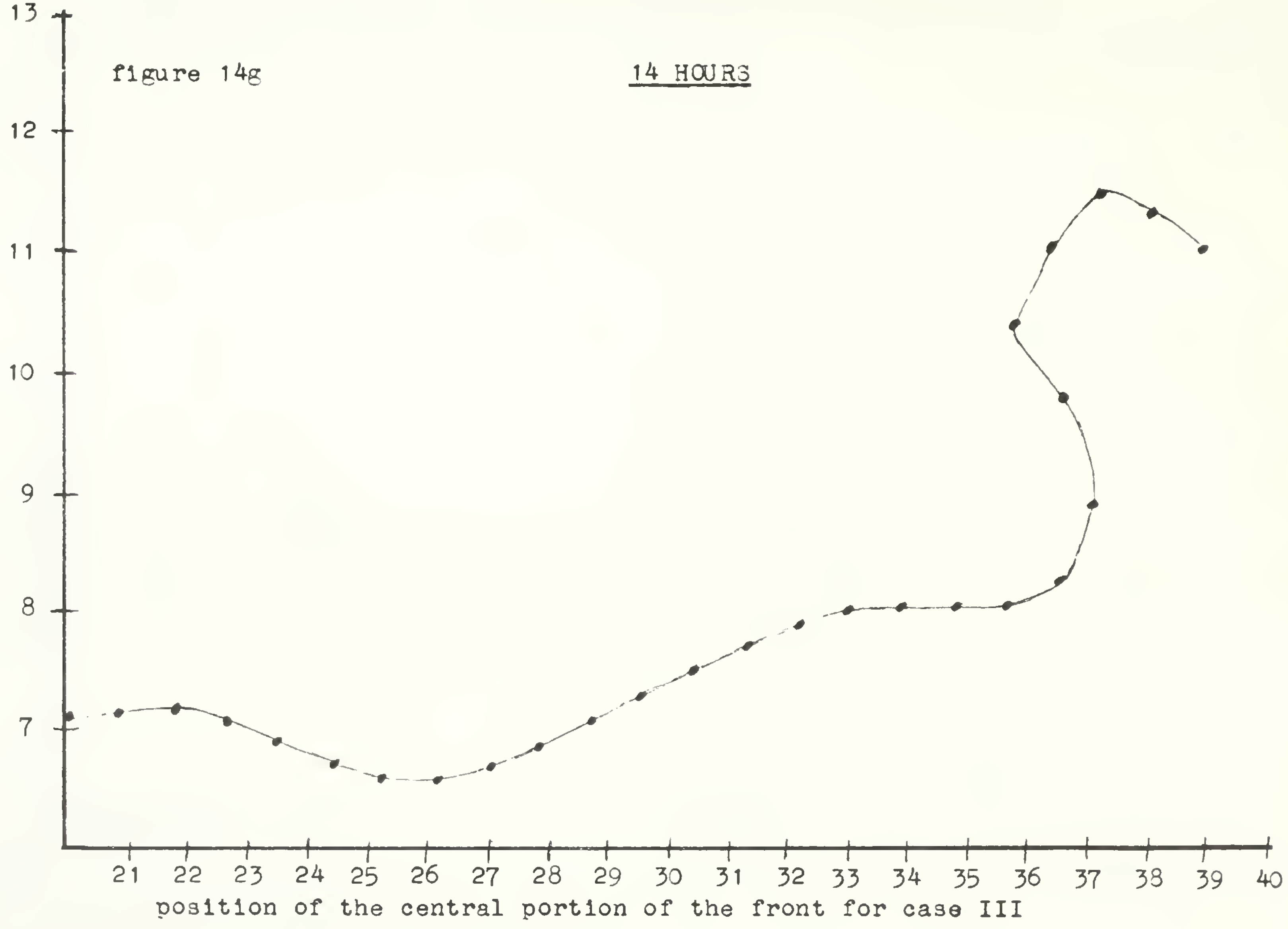

14 HOURS
13
12
11
10
9
8
7
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
position of the central portion of the front for case III

figure 14g

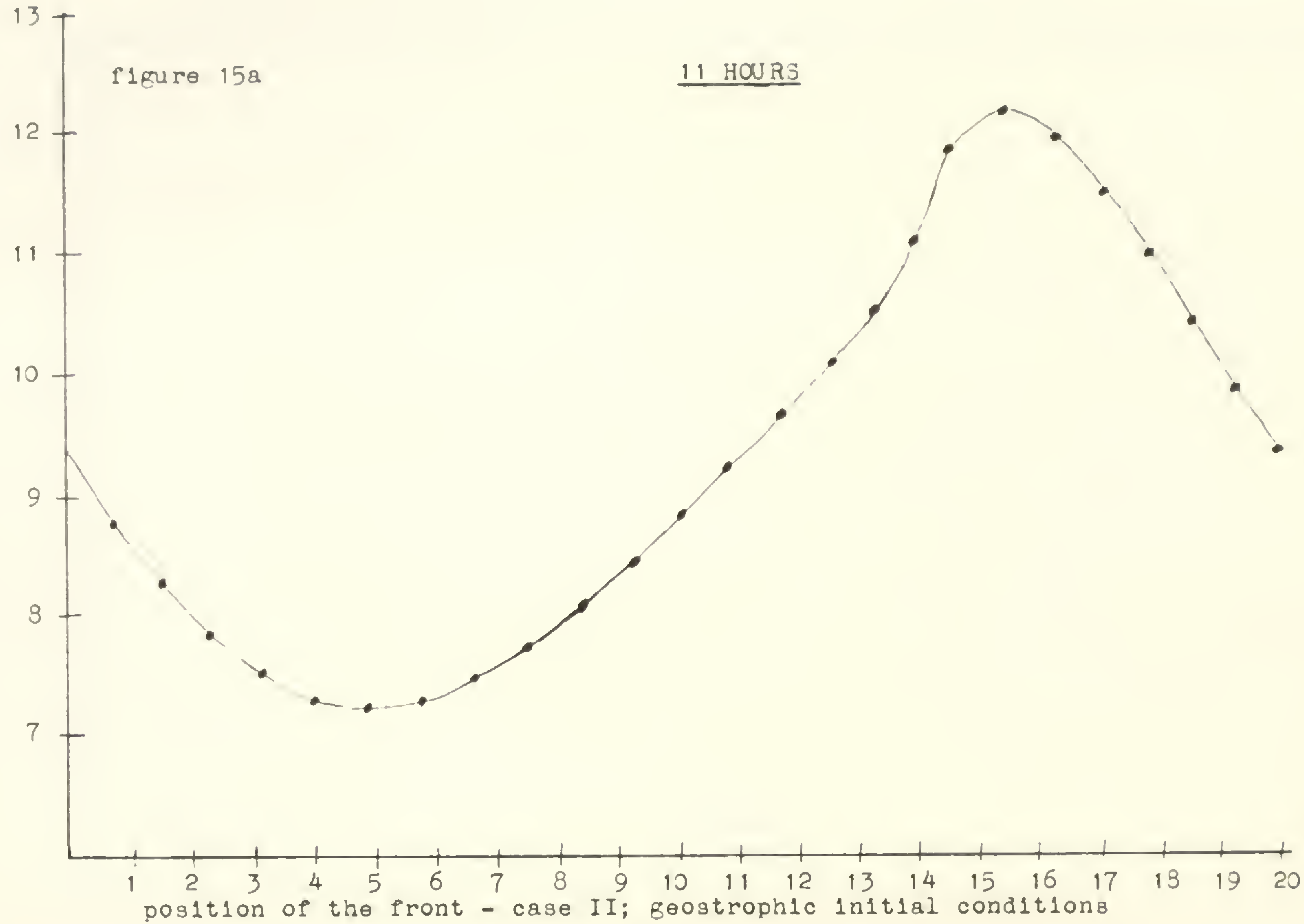
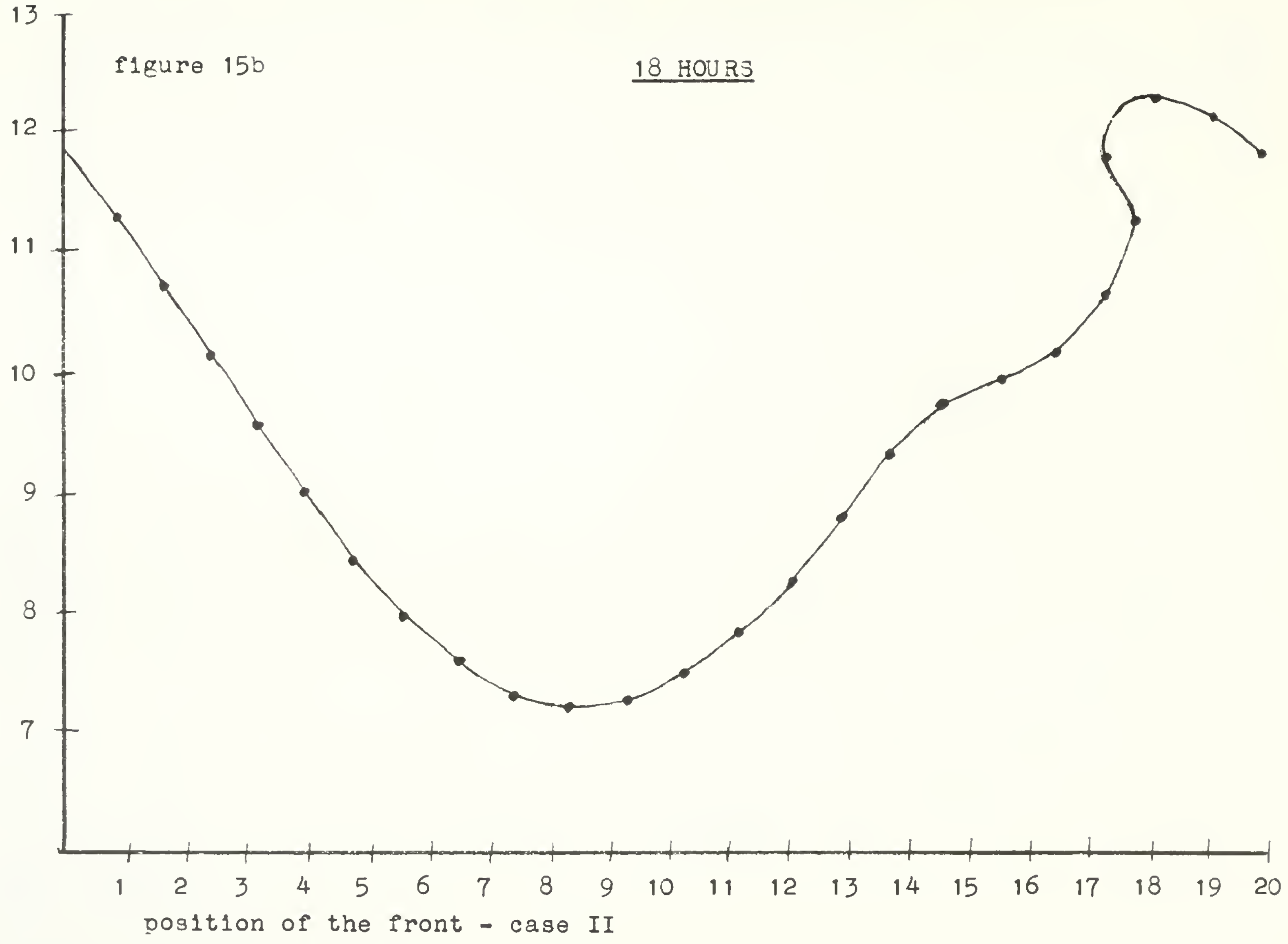


figure 15a
11 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case II; geostrophic initial conditions

figure 15b

18 HOURS



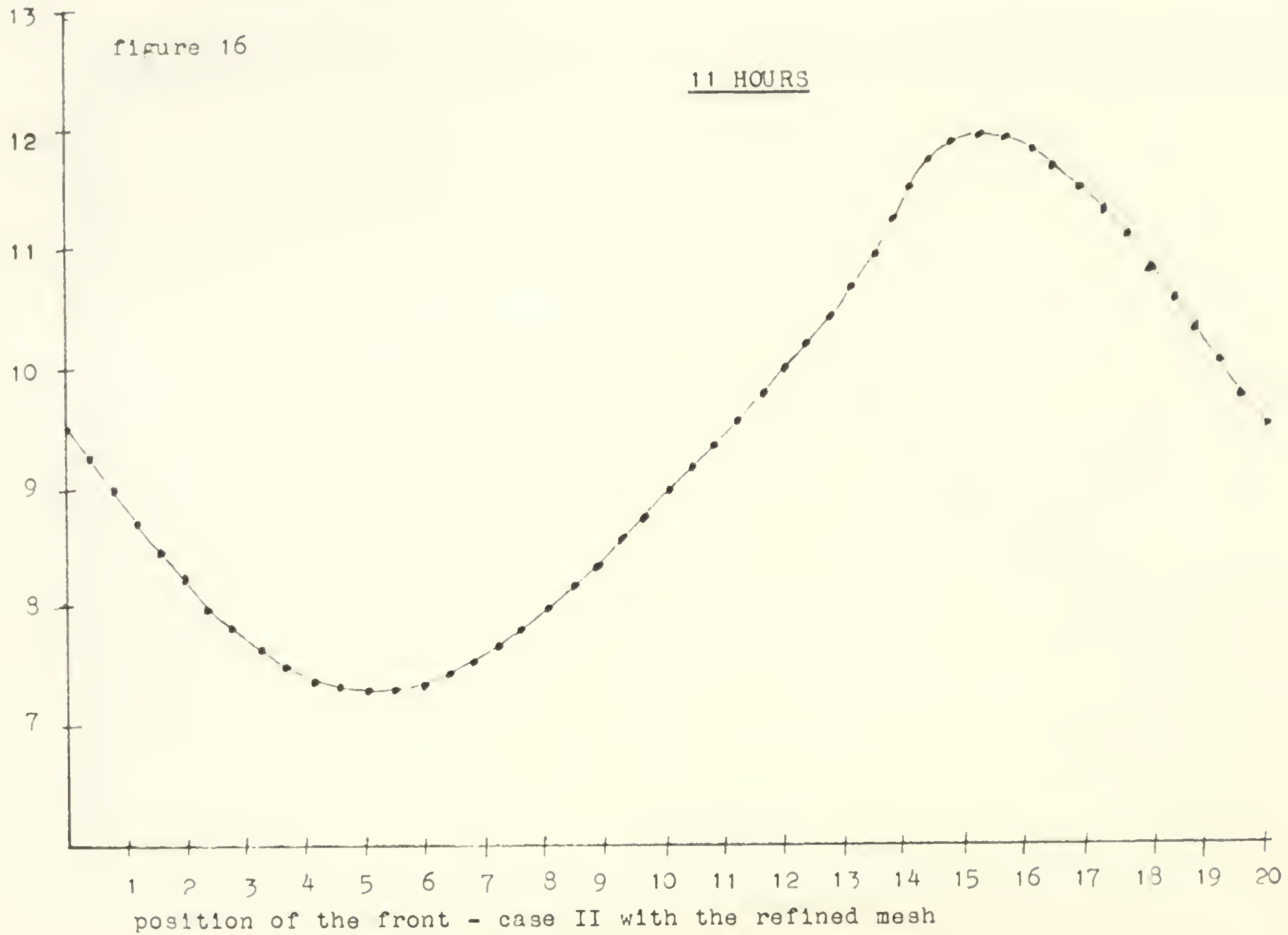


figure 16
11 HOURS
13
12
11
10
9
8
7
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
position of the front - case II with the refined mesh

CHAPTER III. TWO DIMENSIONAL - TWO LAYER THEORY

In this chapter we discuss problem II, as given by equations (1.5). As in the previous chapter the cold air lies above a region D of the x-y plane which is contained within a rectangle R. The domain D is bounded on three sides by straight lines and on the fourth side by the front curve C(t). The warm air lies above the entire rectangle R. Thus, in the domain D we have both the warm and cold air masses. The variables associated with the cold air are denoted by unprimed letters while the warm air variables are denoted by primed letters.

As in the previous chapter we consider a moving coordinate system and introduce dimensionless variables.

$$\tau = \frac{t}{\Delta t} \qquad \lambda = \frac{\Delta t}{\Delta s}$$

$$\xi = \frac{x-\bar{u}t}{\Delta s} \qquad \eta = \frac{y}{\Delta s}$$

$$\hat{u} = \lambda(u-\bar{u}) \qquad \hat{v} = \lambda v$$

$$\hat{h} = \lambda^2 g(1-\frac{\rho'}{\rho})h$$

$$\hat{u}' = \lambda(u'-\bar{u}) \qquad \hat{v} = \lambda v'$$

$$\hat{h}' = \lambda^2 g(1-\frac{\rho'}{\rho})h'$$

$\bar{u}$ is a constant while Δt and Δs are units of time and length. Let

$$F = f\,\Delta t \ , \quad G = -F\lambda\bar{u} \ , \quad r = \frac{1}{1-\frac{\rho'}{\rho}} = \frac{\rho}{\rho-\bar{\rho}} \ .$$

We then use (x,y,t) instead of (ξ,η,τ) and drop the circumflexes. Our system then becomes

$$
(3.1)\qquad
\begin{aligned}
&u_t + uu_x + vu_y + h_x + (r+1)h'_x = Fv\\
&v_t + uv_x + vv_y + h_y + (r+1)h'_y = -Fu + G\\
&h_t + h(u_x+v_y) + uh_x + vh_y = 0\\
&u'_t + u'v'_x + v'u'_y + rh'_x = Fv'\\
&v'_t + u'v'_x + v'v'_y + rh'_y = -Fu' + G\\
&(h'-h)_t + (h'-h)(u'_x+v'_y) + u'(h'-h)_x + v'(h'-h)_y = 0
\end{aligned}
$$

or in vector form

$$
(3.2)\qquad w_t + Aw_x + Bw_y = f
$$

where

$$
w = \begin{pmatrix} u \\ v \\ h \\ u' \\ v' \\ h' \end{pmatrix}
\qquad
f = \begin{pmatrix} Fv \\ -Fu+G \\ 0 \\ Fv' \\ -Fu'+G \\ 0 \end{pmatrix}
= F\begin{pmatrix} v \\ -u \\ 0 \\ v' \\ -u' \\ 0 \end{pmatrix}
+ G\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}
$$

$$
A = \begin{pmatrix}
u & 0 & 1 & 0 & 0 & r+1\\
0 & u & 0 & 0 & 0 & 0\\
h & 0 & u & 0 & 0 & 0\\
0 & 0 & 0 & u' & 0 & r\\
0 & 0 & 0 & 0 & u' & v\\
h & 0 & u-u' & h'-h & 0 & u'
\end{pmatrix}
$$

$$B = \begin{pmatrix} v & 0 & 0 & 0 & 0 & 0 \\ 0 & v & 1 & 0 & 0 & r+1 \\ 0 & h & v & 0 & 0 & 0 \\ 0 & 0 & 0 & v' & 0 & 0 \\ 0 & 0 & 0 & 0 & v' & r \\ 0 & h & v-v' & 0 & h-h' & v' \end{pmatrix}$$

In order for this system to be hyperbolic it is necessary that both A and B have real eigenvalues. Since both matrices A and B have similar structures it is sufficient to analyze A.

$$0 = \det(A - \lambda I)$$

$$= (u-\lambda)(u'-\lambda)\,\{-rh(u'-\lambda)^2 + [(u-\lambda)^2 - h][(u'-\lambda)^2 - r(h'-h)]\}$$

Thus, two of the eigenvalues are $\lambda = u$, $\lambda = u'$. For the other eigenvalues we must solve a fourth order polynomial equation

$$[(u-\lambda)^2 - (r+1)h][(u'-\lambda)^2 - r(h'-h)] = r^2h(h'-h) .$$

Outside the region D there is no cold air and so $h = 0$. Then this equation simplifies and we can solve explicitly for the eigenvalues.

$$\lambda = u \text{ (double)}, \qquad \lambda = u' \pm \sqrt{rh'}$$

In the general case this equation can be solved numerically. It has been found in all the cases treated that the eigenvalues are real when u, u', h, h' are real.

If we compare this system to that obtained when the single layer model is considered we notice several difficulties. First, this system is no longer symmetrizable and so the results of Lax and Wendroff do not necessarily apply even to the linearized equations. Similarly because of the interaction between the warn and cold air masses this system can no longer be converted to conservative form. Thuw we are not able to use the various two step methods employed earlier but one must use one of the more involved one step techniques. In the region where $h = 0$ we calculated that the sound speed is $c' = \sqrt{rh'}$. For the parameters used in the previous chapter r is approxomately 50. If at the southern boundary h is only twice the maximum value of h (i.e. the maximum height of the warm layer is twice the maximum of the cold layer) then $c' \simeq 10c$ where c is the sound speed of the single layer model. Since the size of the time step is inversely proportional to the sound speed we must use time steps that are about 1/10 as large as those in the single layer theory. This together with the necessity for one step method shows that even with modern computers the time required to follow the front for a reasonable length of time is quite large. As an estimate, to follow the front for eight hours of physical time with a coarse 20×20 mesh would require about half an hour of computing time on the CDC 6600. For a 40×40 mesh the time required would jump to about 4 hours. This is compared to about 5 minutes of computing time required to follow the front in the single layer model with the finer mesh.

According to our original assumption the warm layer does not appreciably affect the dynamics of the cold air. Thus, the high sound speeds in the warm air are not physically relevant. So, the small time steps are made necessary for mathematical rather than physical reasons. However, a more difficult problem arises in following the motion of the front. In the single layer theory the front reduces to a curve in the horizontal plane and can be treated as a free boundary, but in the two layer theory the front is a contact discontinuity. Thus, initially the tangential velocities differ on the two sides of the front, also both h and (h'-h) have discontinuous tangents across the front. This discontinuity of the first derivatives probably propagates as a jump discontinuity moving through the air. It is well known that along contact discontinuities Rayleigh and Taylor instabilities can appear. If the underlying differential equations are unstable then the difference approximation must be unstable [15] and hence these physical instabilities are accentuated by the errors inherent in a numerical approximation.

A first attempt to solve this problem was made using as initial conditions the assumptions that were used in constructing the single layer theory. Thus, we assumed that relative to the moving coordinate system $v = v' = 0$, $u = 0$ and $u' = \bar{u}' = \bar{u}$. It was found that instabilities occurred immediately. Subsequently it was found that the situation

improved when the initial conditions were chosen so as to satisfy the jump conditions. Richtmyer [14] also found that for stability it was necessary to insure that the initial conditions satisfied the Hugoniot relationships.

In developing the single layer theory we used the kinematic conditions

$$h_t + u'h_x + v'h_y = 0 \ , \quad h_t + u\, h_x + v\, h_y = 0$$

or after subtracting the first equation from the second,

$$(u-u')h_x + (v-v')h_y = 0 \ .$$

Let w be the velocity vector (u,v) in the cold air and w' the corresponding velocity vector in the warm layer. Then, this equation states that w-w' is perpendicular to ∇h. Since ∇h is normal to the front this equation is a restriction only on the normal component of w-w'. Thus, we have $(w-w')_n \frac{\partial h}{\partial n} = 0$. Since $\frac{\partial h}{\partial n} \neq 0$ we must have $(w-w')_n = 0$, i.e. the normal component of the velocity is continuous across the front. This equation has meaning only as we approach the front from inside the domain D since w is not defined outside of the domain D.

Let $y = y_C(x)$ be the initial location of the front. We then choose as our initial conditions in the moving coordinate system

$$u = 0 \qquad\qquad u' = \bar{u}' - \bar{u}$$

$$v = (\bar{u}-\bar{u}')\frac{dy_C}{dx} \qquad\qquad v' = 0$$

where $\quad y_C = C_1 + C_2 \sin\left(\frac{\pi}{10}\ x\right)$.

With these new initial conditions we were able to continue the solution for several time steps. However instabilities still occurred after about six time steps and before meaningful results could be obtained. Research is continuing to find methods of eliminating these instabilities .

Bibliography

[1] Abdullah, A. J., "Cyclogenesis by a purely mechanical process," Journal of Meteorology, Vol. 6, pp. 86-97 (1949).

[2] Burstein, S., "Finite difference calculations for hydrodynamic flows containing discontinuities," Journal of Computational Physics, Vol. 1, pp. 198-222 (1966).

[3] Burstein, S., "High order accurate difference methods in hydrodynamics," Nonlinear Partial Differential Equations, edited by W. F. Ames, Academic Press, New York, pp. 279-290 (1967).

[4] Ciment, M., "Stable difference schemes with uneven mesh spacings," Courant Inst. Math. Sci., NYU Rep. NYO-1480-100 (1968).

[5] Courant, R., K. O. Friedrichs and H. Lewy, "On the partial difference equations of mathematical physics," Mathematische Annalen Vol. 100, pp. 32-74 (1928). Translation in IBM Journal of Research and Development, Vol. 11, pp. 215-235 (1967).

[6] Freeman, J. C., "Flow under an inversion in middle latitudes," Ph.D. Dissertation, Dept. of Meteorology, University of Chicago (1952).

[7] Houghton, D., A. Kasahara and W. Washington, "Long term integration of the barotropic equations by the Lax-Wendroff scheme," Monthly Weather Review,

Vol. 94, pp. 141-150 (1966).

[8] Kasahara, A., E. Isaacson and J. J. Stoker, "Numerical studies of frontal motion in the atmosphere," Tellus, Vol. 17, pp. 261-276 (1965). (Abbreviated as K.I.S.)

[9] Kasahara, A., E. Isaacson and J. J. Stoker, "Numerical studies of frontal motion in the atmosphere," Courant Inst. Math. Sci., NYU, Report NYO-1480-6 (1964).

[10] Kreiss, H. O., "Difference approximations for the initial-boundary value problem for hyperbolic differential equations," Numerical Solutions of Nonlinear Differential Equations, Edited by D. Greenspan, pp. 141-166 (1966).

[11] Kreiss, H. O., "On difference approximations with wrong boundary values," Mathematics of Computations, Vol. 22, pp. 1-12 (1968).

[12] Lax, P. D., "Hyperbolic systems of conservation laws II," Comm. Pure Appl. Math., Vol. 10, pp. 537-566 (1957).

[13] Lax, P. D. and B. Wendroff, "Difference schemes for hyperbolic equations with high order accuracy," Comm. Pure Appl. Math., Vol. 17, pp. 381-398 (1964).

[14] Richtmyer, R. D., "Progress report on the mach reflection calculation," Courant Inst. Math. Sci., NYU, Report NYO-9764 (1961).

[15] Richtmyer, R. D. and K. W. Morton, Difference methods for initial value problems, Interscience, New York (1967).

[16] Stoker, J. J., "Dynamic theory for treating the motion of cold and warm fronts in the atmosphere," Inst. Math. Sci., NYU, Report IMM-195 (1953).

[17] Stoker, J. J., Water Waves, Interscience, New York (1957).

[18] Tepper, M., "The application of the hydraulic analogy to certain atmospheric flow problems," U. S. Dept. Commerce Weather Bureau Research Paper No. 35 (1952).

[19] Turkel, E., "Frontal motion in the atmosphere," Ph.D. Dissertation, Department of Mathematics, New York University (1970).

[20] Whitham, G. B., "Dynamics of meteorological fronts," Inst. Math. Sci., NYU, Report IMM-195 (1953).

```
      FORTRAN IV PROGRAM FRONUA(INPUT,OUTPUT)
C     TWO DIMENSION TWO LEVEL LAX WENDROFF METHOD
C     BURSTEIN METHOD
      COMMON/A1/AL,AS,AK
      COMMON/A3/AX,AY
      COMMON/A4/F,G
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A10/EPS
      COMMON/A11/IP,IM,JP,JM
      COMMON/A12/K
      COMMON/A15/L(24,27)
      COMMON/A18/ICHT,IPR1,IPR2,IPO,IP1,IP2,JPO
      COMMON/A19/CHPER
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A23/AKT
      COMMON/A24/KT
      COMMON/A25/IPRINT,IPLOT
      COMMON/A26/TT,DT
      COMMON/A30/ITER,ITERT
      COMMON/A36/DISMIN
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A51/FHX(30),FHY(30)
      COMMON/A52/FU(30),FV(30)
      COMMON/PP1/LP
      CALL CONINIT
      IF (IPLOT .GT. 0) CALL PLTINIT
      JPLOT = IABS(IPLOT)
      K = 1
      AKT = 0.
      KT = 0
C     AKT IS PRESENT TIME IN MINUTES
C     KT IS PRESENT NUMBER OF TIME STEPS
      CALL INIT
      IF (IPLOT .GT. 0) CALL MYPLOT
    1 KT = KT+1
      IF (KT .NE. ICHT) GO TO 3
      DT = CHPER*DT
      AK = CHPER*AK
      AL = CHPER*AL
      AS = CHPER*AS
    3 AKT = AKT + DT
      IF (KT .EQ. ICHT) IPRINT = IPC
      IF (KT .EQ. ICHT) JPLOT = JPO
      IF (KT .EQ. IPR1) IPRINT = IP1
      IF (KT .EQ. IPR2) IPRINT = IP2
      KZ = KT-ICHT+9
```

```
      KL = IFTHEN(KT ,GT, ICHT,KZ,KT)
      IPRNOW = MOD(KL,IPRINT)
      IPLNOW = MOD(KL,JPLOT)
      PRINT 900,AKT
      IF (IPRNOW ,EQ, 0) PRINT 901, KT
    9 DO 99 I=3,NI1
      IP = I+1
      IM = I-1
      DO 99 J=1,NJ
      IF (L(I,J) ,LE, 0) GO TO 99
C     L IS EQUAL TO 1 IN THE COLD AIR, 0 IN THE WARM AIR
      JP = J+1
      JM = J-1
      IF (K ,GT, 1) GO TO 20
C     K IS EQUAL TO 2 ON THE SECOND TIME AROUND
C     WE ARE CHECKING IF THE NEAREST NEIGHBORS ARE IN THE COLD AIR
      IF (J ,GE, NJ) GO TO 99
      IF (L(IP,JP),LE,0,OR,L(IP,J),LE,0,OR,L(I,JP),LE,0) GO TO 99
      CALL TIMNEX1(I,J)
C     TIMNEX IS THE TWO STEP BURSTEIN METHOD
      GO TO 99
   20 IF (J-NJ+1) 30,50,70
   30 IF (L(IM,JP) ,LE, 0) GO TO 80
      IF (L(I,JP) ,LE, 0) GO TO 81
      IF (L(IP,JP) ,LE, 0) GO TO 35
C     IF IPOINT=0 THEN NEITHER U(I,JM,3),U(IP,J,M),U(IM,J,M) ARE MISSING
C     IF IPOINT=1 THEN U(I,JM,3) IS MISSING
C     IF IPOINT=2 THEN U(IP,J,3) IS MISSING
C     IF IPOINT=3 THEN U(IM,J,3) IS MISSING
C     IF IPOINT=4 THEN U(I,JM,3) AND U(IP,J,3) ARE MISSING
C     IF IPOINT=5 THEN U(I,JM,3) AND U(IM,J,3) ARE MISSING
      IPOINT = IFTHEN(L(I,JM) ,LE, 0, 1,0)
      IF (L(IP,J) ,LE, 0) IPOINT = 2*IPOINT+2
      IF (L(IM,J) ,GT, 0) GO TO 40
      IF (IPOINT ,GT, 1) GO TO 85
      IPOINT = 2*IPOINT+3
      GO TO 60
   35 IF (L(IM,J) ,LE, 0 ,OR, L(IM,JM) ,LE, 0) GO TO 82
      IPOINT = IFTHEN(L(I,JM) ,LE, 0,7,6)
      PRINT 415, I,J,IPOINT
      GO TO 60
   40 IF (IPOINT,GT,0,OR,L(IP,JM),LE,0,OR,L(IM,JM),LE,0) GO TO 60
   50 CALL TIMNEX2(I,J)
C     TIMNEX IS THE TWO STEP BURSTEIN METHOD
      GO TO 99
   60 CALL ONESTEP(I,J,IPOINT)
      GO TO 99
   70 CALL NOTHBON(I)
```

```
C       NOTHBON IS FOR CALCULATING U,V,H AT THE NORTHERN BOUNDARY
        GO TO 99
   80 IF (LP ,EQ, 0) PRINT 405, I,J,L(I,J),L(IM,JP)
        L(I,J) = 10
        GO TO 99
   81 IF (LP ,EQ, 0) PRINT 406, I,J,L(I,J),L(I,JP)
        L(I,J) = 11
        GO TO 99
   82 IF (LP ,EQ, 0) PRINT 407, I,J,L(I,J),L(IP,JP)
        L(I,J) = 12
        GO TO 99
   85 IF (L(I,JM) ,GT, 0) GO TO 86
        IF (LP ,EQ, 0) PRINT 410, I,J,IPOINT,L(I,J),L(IP,J),L(IM,J)
        L(I,J) = 9
        GO TO 99
   86 IF (LP ,EQ, 0) PRINT 411, I,J
        L(I,J) = 13
   99 CONTINUE
        IF (K ,GT, 1) GO TO 105
        K = 2
        CALL PERIOD(2)
        GO TO 9
  105 K = 1
        ITER = 0
  110 ITER = ITER+1
C       FRONT AND FRONAM CALCULATE THE POSITION OF THE FRONT AT TIME T
        CALL FRONT
        CALL FRONAM
        DO 130 I=3,NI1
        DO 130 J=1,NJ
        CALL POSIN(I,J,IPOS)
        IF (IPOS ,GT, 0) GO TO 120
        L(I,J) = 0
C       0 IS FOR THE WARM AIR AND 1 IS FOR THE COLD AIR
        U(I,J,3) = 0,
        V(I,J,3) = 0,
        H(I,J,3) = 0,
        GO TO 130
  120 IF (L(I,J)-1) 125,130,126
  125 L(I,J) = 2
C       WE ARE CALCULATING U,V,H AT POINTS TOO NEAR TO THE FRONT
  126 CALL TOONER(I,J)
        IF (H(I,J,3) ,GT, 0,) GO TO 130
        IF (ITER ,LE, 1) PRINT 700, H(I,J,3),I,J
        H(I,J,3) = ,00001
  130 CONTINUE
        DO 140 J=1,NJ
        L(1,J) = L(NI,J)
```

```
      L(2,J) = L(NI1,J)
      L(NI2,J) = L(3,J)
  140 L(NI3,J) = L(4,J)
      CALL PERIOD(3)
      CALL FRONFH
C     FRONFH CALCULATES H SUB X AND Y AT THE FRONT
C                       TO BE USED AT THE NEXT FRONT CALCULATION
      IF (ITER .LE. 1) GO TO 145
      IF (ITERT .LE. 0) GO TO 150
C     ITERT = 1 IF WE ARE TO ITERATE AGAIN
      IF (ITER .GT. 10) GO TO 152
  145 ITERT = 0
      GO TO 110
  150 IF (ITER .LE. 4) GO TO 154
  152 PRINT 910, ITER
  154 DO 160 I=3,NI1
      DO 160 J=1,NJ
      IF (L(I,J) .GT. 1) L(I,J) = 1
  160 CONTINUE
      DO 165 J=1,NJ
      L(1,J) = L(NI,J)
      L(2,J) = L(NI1,J)
      L(NI2,J) = L(3,J)
  165 L(NI3,J) = L(4,J)
      IF (LP .EQ. 0 .AND.
     1 (AKT.EQ.600. .OR. AKT.EQ.720. .OR. AKT.EQ.960.)) CALL MYPRNT2(3)
      IF (IPRNOW .EQ. 0) CALL MYPRNT1(3)
      DELS = (FRS(NF3)-FRS(3))/NF
      IF (DISMIN .GT. .75*DELS) GO TO 220
      CALL RELABLE
      CALL FRONFH
      IF (LP .EQ. 0) CALL MYPRNT1(2)
  220 CALL CHGFR
C     CHGFR CHECKS IF FRX IS LESS THAN ZERO OR GREATER THAN NI
      IF (IPLNOW .EQ. 0) CALL MYPLOT
      DO 225 I = 1,NI3
      DO 225 J = 1,NJ
      U(I,J,1) = U(I,J,3)
      V(I,J,1) = V(I,J,3)
  225 H(I,J,1) = H(I,J,3)
      IF (AKT .LT. TT) GO TO 1
      CALL MYEXIT(1)
  405 FORMAT(30H ERROR MESSAGE 1     AT POINT (,I3,1H,,I3,6H) L = ,I2,
     1 12H L(IM,JP) = ,I2/10(1H*))
  406 FORMAT(30H ERROR MESSAGE 2     AT POINT (,I3,1H,,I3,6H) L = ,I2,
     1 11H L(I,JP) = ,I2/10(1H*))
  407 FORMAT(30H ERROR MESSAGE 3     AT POINT (,I3,1H,,I3,6H) L = ,I2,
     1 12H L(IP,JP) = ,I2/10(1H*))
```

```
410 FORMAT(30H ERROR MESSAGE 4     AT POINT (,I3,1H,,I3,14H)     IPOINT
   1= ,I2,5H L = ,I2,11H L(IP,J) = ,I2,11H L(IM,J) = ,I2/10(1H*))
411 FORMAT(30H ERROR MESSAGE 5     AT POINT (,I2,1H,,I3,
   1 73H)  POINTS (IP,J),(IM,J) WERE IN WARM AIR BUT POINT (I,JM) WAS
   2IN COLD AIR)
415 FORMAT(20X,11H AT POINT (,I2,1H,,I2,
   1 34H)  L(IP,JP) WAS MISSING  IPOINT = ,I2)
700 FORMAT(1X,*H IS STILL LESS THAN ZERO AND EQUAL TO *,F12.7,
   1 11H AT POINT (,I2,1H,,I2,1H)/50(1H*))
900 FORMAT(27H THE NUMBER OF MINUTES IS  ,F7.2)
901 FORMAT(29H THE NUMBER OF TIME STEPS IS ,I3)
910 FORMAT(1X,*ITER IS GREATER THAN 3 AND EQUAL TO *,I2)
    END


    SUBROUTINE CONINIT
    COMMON/A1/AL,AS,AK
    COMMON/A3/AX,AY
    COMMON/A4/F,G
    COMMON/A10/EPS
    COMMON/A18/ICHT,IPR1,IPR2,IP0,IP1,IP2,JP0
    COMMON/A19/CHPER
    COMMON/A20/TLEN
    COMMON/A21/NI,NJ,NI1,NI2,NI3
    COMMON/A22/NF,NF1,NF2,NF3,NF4
    COMMON/A25/IPRINT,IPLOT
    COMMON/A26/TT,DT
    COMMON/A31/EPS1
    COMMON/A35/RDIS
    COMMON/A36/DISMIN
    COMMON/PL/SIZEX,SIZEY,TLENY
    COMMON/PR1/LP
    READ 600, IPRINT,IPLOT,LP
    READ 601, TT
    DATA NI,NJ,NF/21,21,25/
    DATA F,G/.18,.05184/
    DATA DT/10./
    DATA AX,AY/1.,1./
    DATA TLEN/20./
    DATA RDIS/.25/
    DATA DISMIN/5./
    DATA CHPER/.75/
    DATA ICHT/55/
    DATA IPR1,IPR2/110,128/
    DATA IP0,IP1,IP2/8,4,1/
    DATA EPS,EPS1/.00001,.000001/
```

```
      DATA SIZPX,SIZEY,TLENY/7.874,2.75,13./
      AK = 1./3.
      AL = AK/AX
      AS = AK/AY
      JPO = IFTHEN(IPLOT .GT. 0,8,1000)
      AK IS DELTA T  AX IS DELTA X  AY IS DELTA Y
      TT IS THE TOTAL TIME IN MINUTES
      DT IS THE TIME STEP IN MINUTES
      NI IS THE TOTAL NUMBER OF POINTS IN THE X DIRECTION
      NJ IS THE TOTAL NUMBER OF POINTS IN THE Y DIRECTION
      NF IS THE TOTAL NUMBER OF FRONT POINTS
      IF LP IS NONZERO WE ELIMINATE MOST OF THE PRINTOUT
      IPRINT IS THE NUMBER OF TIME STEPS BETWEEN PRINTOUTS
      IPLOT IS THE NUMBER OF TIME STEPS BETWEEN PLOTS
      TLEN IS THE LENGTH OF THE X AXIS
      IF A POINT IS LESS THAN RDIS TO THE BOUNDARY WE USE INTERPOLATION
                    INSTEAD OF THE DIFFERENTIAL EQUATIONS
      CHPER IS THE PERCENTAGE WE REDUCE AK WHEN WE VIOLATE STABILITY
      ICHT IS THE TIME (KT) WHEN WE REDUCE AK BECAUSE OF STABILITY NEEDS
      IPR1, IPR2 ARE THE TIME STEPS WHEN WE BEGIN PRINTING MORE OFTEN
      IPO,IP1,IP2 ARE HOW OFTEN WE PRINT AT TIMES ICHT,IPR1,IPR2
      JPO IS HOW OFTEN WE PLOT AFTER TIME ICHT
      EPS IS THE ERROR ALLOWED IN THE SOLUTION OF FRX(XX)=AJ
      EPS1 IS THE CHANGE ALLOWED BETWEEN ITERATES IN SOLVING
                    THE O.D.E. TO MOVE THE FRONT
      SIZE IS THE LENGTH OF THE AXIS TO BE PLOTTED  (IN INCHES)
      TLENY IS THE LENGTH OF THE Y COORDINATE X IN TERMS OF DELTA Y
      NI1 = NI+1
      NI2 = NI+2
      NI3 = NI+3
      NF1 = NF+1
      NF2 = NF+2
      NF3 = NF+3
      NF4 = NF+4
      RETURN
  600 FORMAT(3I5)
  601 FORMAT(F10.2)
      END




      SUBROUTINE PLTINIT
      CALL PLOTS(250,20HI.U. 12910B TURKEL   )
      CALL PLOT(0.,-11.,-3)
      CALL PLOT(1.,1.,-3)
      RETURN
      END
```

```
      SUBROUTINE INIT
      COMMON/A3/AX,AY
      COMMON/A4/F,G
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A15/L(24,27)
      COMMON/A20/TLEN
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A51/FHX(30),FHY(30)
      COMMON/A52/FU(30),FV(30)
      PIT = 2.*3.141592653/TLEN
      DO 1 I=3,NF2
      AI = (I-3)*AX
      FRX(I) = .8*AI
      FRY(I) = 9.5*TLEN/20.-.1*TLEN*COS(PIT*FRX(I))
      FHX(I) = -.1*TLEN*PIT*G*SIN(PIT*FRX(I))
      FHY(I) = G
      FU(I) = 0.
    1 FV(I) = 0.
      CALL PER2(1)
      FRS(2) = 0.
      DO 2 I=3,NF3
      CALL FRONDIS(I,0.,0.,DIFDIS)
    2 FRS(I) = FRS(I-1)+DIFDIS
      CALL PER2(-1)
      CALL FRONPO
C     FRX IS THE X COORDINATE OF THE I-TH POINT
C     FHX IS H SUB X AT THE FRONT POINTS
C     FU IS U AT THE FRONT POINTS
      DO 8 I=3,NI1
      AI = (I-3)*AX
      FF(I) = 9.5*TLEN/20.-.1*TLEN*COS(PIT*AI)
C     FF IS THE POSITION OF THE FRONT AT COORDINATE LINES
      DO 8 J=1,NJ
      AJ = (J-1)*AY
      U(I,J,1) = 0.
      V(I,J,1) = 0.
      IF (AJ .LE. FF(I)) GO TO 7
      H(I,J,1) = G*(AJ-FF(I))
      L(I,J) = 1
C     IF L = 1 THEN THE POINT IS IN THE COLD AIR
```

```
      GO TO 8
    7 H(I,J,1) = 0.
      L(I,J) = 0
      IF L = 0 THEN THE POINT IS IN THE WARM AIR
    8 CONTINUE
      CALL PERIOD(1)
      DO 10 J=1,NJ
      L(1,J) = L(NI,J)
      L(2,J) = L(NI1,J)
      L(NI2,J) = L(3,J)
   10 L(NI3,J) = L(4,J)
      DO 20 I=1,NI3
      DO 20 J=1,NJ
      DO 20 M=2,3
      U(I,J,M) = U(I,J,1)
      V(I,J,M) = V(I,J,1)
   20 H(I,J,M) = H(I,J,1)
      CALL MYPRNT1(2)
      RETURN
      END




      SUBROUTINE TIMNEX1(I,J)
      TIMNEX1 IS THE FIRST STEP OF THE TWO-STEP BURSTEIN METHOD
      COMMON/A1/AL,AS,AK
      COMMON/A4/F,G
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A11/IP,IM,JP,JM
      (I,J,2) REPRESENTS (I + 1/2,J + 1/2,2)
      H(I,J,2)=.25*(H(IP,JP,1)+H(IP,J,1)+H(I,JP,1)+H(I,J,1))
     1-.5*AL*(H(IP,JP,1)*U(IP,JP,1)-H(I,JP,1)*U(I,JP,1)
     2+H(IP,J,1)*U(IP,J,1)-H(I,J,1)*U(I,J,1))
     3-.5*AS*(H(IP,JP,1)*V(IP,JP,1)-H(IP,J,1)*V(IP,J,1)
     4+H(I,JP,1)*V(I,JP,1)-H(I,J,1)*V(I,J,1))
      U(I,J,2)=(.25*(H(IP,JP,1)*U(IP,JP,1)+H(IP,J,1)*U(IP,J,1)
     1+H(I,JP,1)*U(I,JP,1)+H(I,J,1)*U(I,J,1))
     2-.5*AL*(H(IP,JP,1)*U(IP,JP,1)*U(IP,JP,1)
     3-H(I,JP,1)*U(I,JP,1)*U(I,JP,1)+H(IP,J,1)*J(IP,J,1)*U(IP,J,1)
     4-H(I,J,1)*U(I,J,1)*U(I,J,1)
     5+.5*(H(IP,JP,1)*H(IP,JP,1)-H(I,JP,1)*H(I,JP,1)
     6+H(IP,J,1)*H(IP,J,1)-H(I,J,1)*H(I,J,1)))
     7-.5*AS*(H(IP,JP,1)*U(IP,JP,1)*V(IP,JP,1)
     8-H(IP,J,1)*U(IP,J,1)*V(IP,J,1)+H(I,JP,1)*J(I,JP,1)*V(I,JP,1)
     9-H(I,J,1)*U(I,J,1)*V(I,J,1))
     A+.25*AK*F*(H(IP,JP,1)*V(IP,JP,1)+H(IP,J,1)*V(IP,J,1)
```

```
     8+H(I,JP,1)*V(I,JP,1)+H(I,J,1)*V(I,J,1)))/H(I,J,2)
      V(I,J,2)=(.25*(H(IP,JP,1)*V(IP,JP,1)+H(IP,J,1)*V(IP,J,1)
     1+H(I,JP,1)*V(I,JP,1)+H(I,J,1)*V(I,J,1))
     2-.5*AL*(H(IP,JP,1)*U(IP,JP,1)*V(IP,JP,1)
     3-H(I,JP,1)*U(I,JP,1)*V(I,JP,1)+H(IP,J,1)*U(IP,J,1)*V(IP,J,1)
     4-H(I,J,1)*U(I,J,1)*V(I,J,1))
     5-.5*AS*(H(IP,JP,1)*V(IP,JP,1)*V(IP,JP,1)
     6-H(IP,J,1)*V(IP,J,1)*V(IP,J,1)+H(I,JP,1)*V(I,JP,1)*V(I,JP,1)
     7-H(I,J,1)*V(I,J,1)*V(I,J,1)
     8+.5*(H(IP,JP,1)*H(IP,JP,1)-H(IP,J,1)*H(IP,J,1)
     9+H(I,JP,1)*H(I,JP,1)-H(I,J,1)*H(I,J,1)))
     A+.25*AK*(G*(H(IP,JP,1)+H(IP,J,1)+H(I,JP,1)+H(I,J,1))
     B-F*(H(IP,JP,1)*U(IP,JP,1)+H(IP,J,1)*U(IP,J,1)
     C+H(I,JP,1)*U(I,JP,1)+H(I,J,1)*U(I,J,1)))))/H(I,J,2)
      RETURN
      END



      SUBROUTINE TIMNEX2(I,J)
C     TIMNEX2 IS THE SECOND STEP OF THE TWO-STEP BURSTEIN METHOD
      COMMON/A1/AL,AS,AK
      COMMON/A4/F,G
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A11/IP,IM,JP,JM
C     (I,J,2)   REPRESENTS  (I + 1/2,J + 1/2,2)
C     (IM,J,2) REPRESENTS  (I - 1/2,J + 1/2,2)
C     (I,JM,2) REPRESENTS  (I + 1/2,J - 1/2,2)
C     (IM,JM,2) REPRESENTS (I - 1/2,J - 1/2,2)
      H(I,J,3) = H(I,J,1)
     1-.25*AL*(H(IP,J,1)*U(IP,J,1)-H(IM,J,1)*U(IM,J,1)
     2+H(I,J,2)*U(I,J,2)-H(IM,J,2)*U(IM,J,2)
     3+H(I,JM,2)*U(I,JM,2)-H(IM,JM,2)*U(IM,JM,2))
     4-.25*AS*(H(I,JP,1)*V(I,JP,1)-H(I,JM,1)*V(I,JM,1)
     5+H(I,J,2)*V(I,J,2)-H(I,JM,2)*V(I,JM,2)
     6+H(IM,J,2)*V(IM,J,2)-H(IM,JM,2)*V(IM,JM,2))
      U(I,J,3)=(H(I,J,1)*U(I,J,1)
     1-.25*AL*(H(IP,J,1)*U(IP,J,1)*U(IP,J,1)
     2-H(IM,J,1)*U(IM,J,1)*U(IM,J,1)+H(I,J,2)*U(I,J,2)*U(I,J,2)
     3-H(IM,J,2)*U(IM,J,2)*U(IM,J,2)+H(I,JM,2)*U(I,JM,2)*U(I,JM,2)
     4-H(IM,JM,2)*U(IM,JM,2)*U(IM,JM,2)
     5+.5*(H(IP,J,1)*H(IP,J,1)-H(IM,J,1)*H(IM,J,1)
     6+H(I,J,2)*H(I,J,2)-H(IM,J,2)*H(IM,J,2)
     7+H(I,JM,2)*H(I,JM,2)-H(IM,JM,2)*H(IM,JM,2)))
     8-.25*AS*(H(I,JP,1)*U(I,JP,1)*V(I,JP,1)
     9-H(I,JM,1)*U(I,JM,1)*V(I,JM,1)+H(I,J,2)*U(I,J,2)*V(I,J,2)
```

```
     A=H(I,JM,2)*U(I,JM,2)*V(I,JM,2)+H(IM,J,2)*U(IM,J,2)*V(IM,J,2)
     B=H(IM,JM,2)*U(IM,JM,2)*V(IM,JM,2))
     C+,5*AK*F*(H(I,J,1)*V(I,J,1)
     D+,25*(H(I,J,2)*V(I,J,2)+H(I,JM,2)*V(I,JM,2)
     E+H(IM,J,2)*V(IM,J,2)+H(IM,JM,2)*V(IM,JM,2))))/H(I,J,3)
      V(I,J,3)=(H(I,J,1)*V(I,J,1)
     1=,25*AL*(H(IP,J,1)*U(IP,J,1)*V(IP,J,1)
     2=H(IM,J,1)*U(IM,J,1)*V(IM,J,1)+H(I,J,2)*U(I,J,2)*V(I,J,2)
     3=H(IM,J,2)*U(IM,J,2)*V(IM,J,2)+H(I,JM,2)*U(I,JM,2)*V(I,JM,2)
     4=H(IM,JM,2)*U(IM,JM,2)*V(IM,JM,2))
     5=,25*AS*(H(I,JP,1)*V(I,JP,1)*V(I,JP,1)
     6=H(I,JM,1)*V(I,JM,1)*V(I,JM,1)+H(I,J,2)*V(I,J,2)*V(I,J,2)
     7=H(I,JM,2)*V(I,JM,2)*V(I,JM,2)+H(IM,J,2)*V(IM,J,2)*V(IM,J,2)
     8=H(IM,JM,2)*V(IM,JM,2)*V(IM,JM,2)
     9+,5*(H(I,JP,1)*H(I,JP,1)-H(I,JM,1)*H(I,JM,1)
     A+H(I,J,2)*H(I,J,2)-H(I,JM,2)*H(I,JM,2)
     B+H(IM,J,2)*H(IM,J,2)-H(IM,JM,2)*H(IM,JM,2)))
     C+,5*AK*(G*(H(I,J,1)
     D+,25*(H(I,J,2)+H(I,JM,2)+H(IM,J,2)+H(IM,JM,2)))
     E-F*(H(I,J,1)*U(I,J,1)
     F+,25*(H(I,J,2)*U(I,J,2)+H(I,JM,2)*U(I,JM,2)
     G+H(IM,J,2)*U(IM,J,2)+H(IM,JM,2)*U(IM,JM,2))))))/H(I,J,3)
      SB = AL*(SQRT(U(I,J,3)*U(I,J,3)+V(I,J,3)*V(I,J,3))+SQRT(H(I,J,3)))
      IF (SB .GE. .5) PRINT 750, I,J,SB,U(I,J,3),V(I,J,3),H(I,J,3)
      RETURN
750   FORMAT(11H AT POINT (,I2,1H,,I2,10H)  STAB = ,F9.5,
     1 6H  U = ,F9.5,6H  V = ,F9.5,6H  H = ,F9.5,33X,
     2 17HSUBROUTINE TIMNEX)
      END



      SUBROUTINE ONESTEP(I,J,IPOINT)
      ONESTEP IS FOR POINTS NEAR THE FRONT
              WHERE UNSYMMETRIC DIFFERENCES ARE REQUIRED
         AND USES AT MOST THE POINTS (IP,JP),(I,JP),(IM,JP),(IP,J),(I,J)
      REAL K1,K2
      COMMON/A1/AL,AS,AK
      COMMON/A3/AX,AY
      COMMON/A4/F,G
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A11/IP,IM,JP,JM
      COMMON/A15/L(24,27)
      COMMON/A35/RDIS
      COMMON/A40/FF(25,6)
      M = 1
```

```
      AI = (I-3)*AX
      AJ = (J-1)*AY
      DAX = 1./AX
      DAY = 1./AY
      DELX = AX
      IF IPOINT=-1 THEN WE HAVE ALL 8 OF THE NEAREST NEIGHBORS
      IF IPOINT=0 THEN NEITHER U(I,JM,3),U(IP,J,M),U(IM,J,M) ARE MISSING
      IF IPOINT=1 THEN U(I,JM,3) IS MISSING
      IF IPOINT=2 THEN U(IP,J,3) IS MISSING
      IF IPOINT=3 THEN U(IM,J,3) IS MISSING
      IF IPOINT=4 THEN U(I,JM,3) AND U(IP,J,3) ARE MISSING
      IF IPOINT=5 THEN U(I,JM,3) AND U(IM,J,3) ARE MISSING
      IF (IPOINT .LE. 1) GO TO 100
      WE ARE TRYING TO FIND THE NEAREST FRONT POINT TO COLUMN AI
      UVMIS CALCULATES U AND V ON THE FRONT
      GO TO (100,200,300,200,300,200,200), IPOINT
      USX IS U SUB X        UXX IS U SUB XX        UXY IS U SUB XY
      USY IS U SUB Y       UYY IS U SUB YY
      UT IS U SUB T   UXT IS U SUB XT   UYT IS U SUB YT   UTT IS U SUB T
      SECTION 100 IS FOR WHEN NEITHER (IP,J) NOR (IM,J) IS MISSING
                 I.E  WHEN IPOINT = 0,1
  100 USX = (U(IP,J,M)-U(IM,J,M))/(2.*AX)
      VSX = (V(IP,J,M)-V(IM,J,M))/(2.*AX)
      HSX = (H(IP,J,M)-H(IM,J,M))/(2.*AX)
      UXX = (U(IP,J,M)-2.*U(I,J,M)+U(IM,J,M))/(AX*AX)
      VXX = (V(IP,J,M)-2.*V(I,J,M)+V(IM,J,M))/(AX*AX)
      HXX = (H(IP,J,M)-2.*H(I,J,M)+H(IM,J,M))/(AX*AX)
      IF (IPOINT) 400,400,500
      SECTION 200 IS FOR WHEN (IP,J) IS MISSING    I.E  IPOINT = 2,4
  200 CALL NEREST(AI,AJ,1,KB)
      CALL UVMISX(I,J,1,KB,UFX,VFX,FX)
      UVMIS CALCULATES U AND V ON THE FRONT
      K1 = FX
      K2 = AX
      DELX = FX
      IF (K1 .LT. RDIS*AX) GO TO 999
      DK1 = 1./(K1*(K1+K2))
      DK2 = 1./(K1*K2)
      DK3 = 1./(K2*(K1+K2))
      USX = K2*DK1*UFX+(K1-K2)*DK2*U(I,J,M)-K1*DK3*U(IM,J,M)
      VSX = K2*DK1*VFX+(K1-K2)*DK2*V(I,J,M)-K1*DK3*V(IM,J,M)
      HSX = (K1-K2)*DK2*H(I,J,M)-K1*DK3*H(IM,J,M)
      UXX = 2.*(DK1*UFX-DK2*U(I,J,M)+DK3*U(IM,J,M))
      VXX = 2.*(DK1*VFX-DK2*V(I,J,M)+DK3*V(IM,J,M))
      HXX = 2.*(-DK2*H(I,J,M)+DK3*H(IM,J,M))
      GO TO (400,400,400,500,500,400,500), IPOINT
      SECTION 300 IS FOR WHEN (IM,J) IS MISSING    I.E  IPOINT = 3,5
  300 CALL NEREST(AI,AJ,-1,KB)
```

```
      CALL UVMISX(I,J,-1,KB,UFX,VFX,FX)
      K1 = AX
      K2 = FX
      DELX = FX
      IF (K2 .LT. RDIS*AX) GO TO 999
      DK1 = 1./(K1*(K1+K2))
      DK2 = 1./(K1*K2)
      DK3 = 1./(K2*(K1+K2))
      USX = K2*DK1*U(IP,J,M)+(K1-K2)*DK2*U(I,J,M)-K1*DK3*UFX
      VSX = K2*DK1*V(IP,J,M)+(K1-K2)*DK2*V(I,J,M)-K1*DK3*VFX
      HSX = K2*DK1*H(IP,J,M)+(K1-K2)*DK2*H(I,J,M)
      UXX = 2.*(DK1*U(IP,J,M)-DK2*U(I,J,M)+DK3*UFX)
      VXX = 2.*(DK1*V(IP,J,M)-DK2*V(I,J,M)+DK3*VFX)
      HXX = 2.*(DK1*H(IP,J,M)-DK2*H(I,J,M))
      IF (IPOINT-4) 400,400,500
C     SECTION 400 IS FOR WHEN (I,JM) IS NOT MISSING I.E. IPOINT=0,2,3
  400 USY = (U(I,JP,M)-U(I,JM,M))/(2.*AY)
      VSY = (V(I,JP,M)-V(I,JM,M))/(2.*AY)
      HSY = (H(I,JP,M)-H(I,JM,M))/(2.*AY)
      UYY = (U(I,JP,M)-2.*U(I,J,M)+U(I,JM,M))/(AY*AY)
      VYY = (V(I,JP,M)-2.*V(I,J,M)+V(I,JM,M))/(AY*AY)
      HYY = (H(I,JP,M)-2.*H(I,J,M)+H(I,JM,M))/(AY*AY)
      GO TO 600
C     SECTION 500 IS FOR WHEN (I,JM) IS MISSING     I.E  IPOINT = 1,4,5
  500 CALL NERESTY(I,AJ,-1,KB)
      CALL UVMISY(I,J,KB,UFY,VFY)
      K1 = AY
      K2 = AJ-FF(I)
      IF (K2 .LT. RDIS*AY) GO TO 999
      DK1 = 1./(K1*(K1+K2))
      DK2 = 1./(K1*K2)
      DK3 = 1./(K2*(K1+K2))
      USY = K2*DK1*U(I,JP,M)+(K1-K2)*DK2*U(I,J,M)-K1*DK3*UFY
      VSY = K2*DK1*V(I,JP,M)+(K1-K2)*DK2*V(I,J,M)-K1*DK3*VFY
      HSY = K2*DK1*H(I,JP,M)+(K1-K2)*DK2*H(I,J,M)
      UYY = 2.*(DK1*U(I,JP,M)-DK2*U(I,J,M)+DK3*UFY)
      VYY = 2.*(DK1*V(I,JP,M)-DK2*V(I,J,M)+DK3*VFY)
      HYY = 2.*(DK1*H(I,JP,M)-DK2*H(I,J,M))
      IF (DELX-K2) 600,600,550
  550 DELX = K2
  600 IF (IPOINT) 700,601,601
  601 IF (IPOINT .GE. 6) GO TO 602
      UXY = (.5*(U(IP,JP,M)-U(IM,JP,M))/AX-USX)/AY
      VXY = (.5*(V(IP,JP,M)-V(IM,JP,M))/AX-VSX)/AY
      HXY = (.5*(H(IP,JP,M)-H(IM,JP,M))/AX-HSX)/AY
      GO TO 800
  602 UXY = (USY-.5*(U(IM,JP,M)-U(IM,JM,M))*DAY)*DAX
      VXY = (VSY-.5*(V(IM,JP,M)-V(IM,JM,M))*DAY)*DAX
```

```
      HXY = (HSY-.5*(H(IM,JP,M)-H(IM,JM,M))*DAY)*DAX
      GO TO 800
C     SECTION 700 IS FOR WHEN ALL 8 NEAREST NEIGHBORS ARE PRESENT
C                I.E WHEN IPOINT = -1
  700 UXY = .25*(U(IP,JP,M)-U(IM,JP,M)-U(IP,JM,M)+U(IM,JM,M))/(AX*AY)
      VXY = .25*(V(IP,JP,M)-V(IM,JP,M)-V(IP,JM,M)+V(IM,JM,M))/(AX*AY)
      HXY = .25*(H(IP,JP,M)-H(IM,JP,M)-H(IP,JM,M)+H(IM,JM,M))/(AX*AY)
  800 UU = U(I,J,M)
      VV = V(I,J,M)
      HH = H(I,J,M)
      UT = -UU*USX-VV*USY-HSX+F*VV
      VT = -UU*VSX-VV*VSY-HSY-F*UU+G
      HT = -HH*(USX+VSY)-UU*HSX-VV*HSY
      UXT = -USX*USX-UU*UXX-VSX*USY-VV*UXY-HXX+F*VSX
      UYT = -USX*USY-UU*UXY-VSY*USY-VV*UYY-HXY+F*VSY
      VXT = -USX*VSX-UU*VXX-VSX*VSY-VV*VXY-HXY-F*USX
      VYT = -USY*VSX-UU*VXY-VSY*VSY-VV*VYY-HYY-F*USY
      HXT = -HSX*(USX+VSY)-HH*(UXX+VXY)-USX*HSX-UU*HXX-VSX*HSY-VV*HXY
      HYT = -HSY*(USX+VSY)-HH*(UXY+VYY)-USY*HSX-UU*HXY-VSY*HSY-VV*HYY
      UTT = -UT*USX-UU*UXT-VT*USY-VV*UYT-HXT+F*VT
      VTT = -UT*VSX-UU*VXT-VT*VSY-VV*VYT-HYT-F*UT
      HTT = -HT*(USX+VSY)-HH*(UXT+VYT)-UT*HSX-UU*HXT-VT*HSY-VV*HYT
      U(I,J,3) = U(I,J,M)+AK*UT+.5*AK*AK*UTT
      V(I,J,3) = V(I,J,M)+AK*VT+.5*AK*AK*VTT
      H(I,J,3) = H(I,J,M)+AK*HT+.5*AK*AK*HTT
      IF (H(I,J,3) .LT. 0.) GO TO 850
      AU = ABS(U(I,J,3))
      AV = ABS(V(I,J,3))
      UL = THENIF(AU .GT. AV,AU,AV)
      SB = AK*(UL+SQRT(H(I,J,3)))/DELX
      IF (SB .GE. .5) PRINT 950, I,J,SB,U(I,J,3),V(I,J,3),H(I,J,3),DELX
      RETURN
  850 H(I,J,3) = 0.
      PRINT 960, I,J,H(I,J,3)
  999 L(I,J) = IPOINT+1
      RETURN
  950 FORMAT(11H AT POINT (,I2,1H,,I2,10H)  STAB = ,F9.5,
     1 6H  U = ,F9.5,6H  V = ,F9.5,6H  H = ,F9.5,8H DELX = ,F9.5,15X,
     2 18HSUBROUTINE ONESTEP)
  960 FORMAT(11H AT POINT (,I2,1H,,I2,47H) H WAS LESS THAN ZERO IN ONEST
     1EP AND EQUAL TO ,F12.7/100(1H*))
      END



      SUBROUTINE NOTHBON(I)
C     NOTHBON IS USED FOR POINTS ON THE NORTHERN BOUNDARY
```

```
C         AT THE BOUNDARY WE USE ONE SIDEL DIFFERENCES
          COMMON/A1/AL,AS,AK
          COMMON/A3/AX,AY
          COMMON/A4/F,G
          COMMON/A5/U(24,27,3),V(24,27,3)
          COMMON/A6/H(24,27,3)
          COMMON/A11/IP,IM,JP,JM
          COMMON/A21/NI,NJ,NI1,NI2,NI3
          J = NJ
          DAX = 1./AX
          DAY = 1./AY
          M = 1
          UU = U(I,J,M)
          VV = V(I,J,M)
          HH = H(I,J,M)
          USX = .5*(U(IP,J,M)-U(IM,J,M))*DAX
          HSX = .5*(H(IP,J,M)-H(IM,J,M))*DAX
          VSX = 0.
          USY = .5*(U(I,J-2,M)-4.*U(I,JM,M)+3.*U(I,J,M))*DAY
          HSY = .5*(H(I,J-2,M)-4.*H(I,JM,M)+3.*H(I,J,M))*DAY
          VSY = .5*(V(I,J-2,M)-4.*V(I,JM,M))*DAY
          UXX = (U(IP,J,M)-2.*U(I,J,M)+U(IM,J,M))*DAX**2
          HXX = (H(IP,J,M)-2.*H(I,J,M)+H(IM,J,M))*DAY**2
          VXX = 0.
          UYY = (U(I,J-2,M)-2.*U(I,JM,M)+U(I,J,M))*DAY**2
          HYY = (H(I,J-2,M)-2.*H(I,JM,M)+H(I,J,M))*DAY**2
          VYY = (V(I,J-2,M)-2.*V(I,JM,M))*DAY**2
          UXY = .5*(U(IP,J,M)-U(IP,JM,M)-U(IM,J,M)+U(IM,JM,M))*DAX*DAY
          HXY = .5*(H(IP,J,M)-H(IP,JM,M)-H(IM,J,M)+H(IM,JM,M))*DAX*DAY
          VXY = -.5*(V(IP,JM,M)-V(IM,JM,M))*DAX*DAY
          UT = -UU*USX-HSX+F*VV
          VT = -UU*VSX-HSY-F*UU+G
          HT = -HH*(USX+VSY)-UU*HSX
          UXT = -USX*USX-UU*UXX-VSX*USY-HXX+F*VSX
          UYT = -USX*USY-UU*UXY-VSY*USY-HXY+F*VSY
          VXT = -USX*VSX-UU*VXX-VSX*VSY-HXY-F*USX
          VYT = -USY*VSX-UU*VXY-VSY*VSY-HYY-F*USY
          HXT = -HSX*(USX+VSY)-HH*(UXX+VXY)-USX*HSX-UU*HXX-VSX*HSY
          HYT = -HSY*(USX+VSY)-HH*(UXY+VYY)-USY*HSX-UU*HXY-VSY*HSY
          UTT = -UT*USX-UU*UXT-VT*USY-HXT-F*VT
          VTT = -UT*VSX-UU*VXT-VT*VSY-HYT-F*UT
          HTT = -HT*(USX+VSY)-HH*(UXT+VYT)-UT*HSX-UU*HXT-VT*HSY
          U(I,J,3) = U(I,J,M)+AK*UT+.5*AK*AK*UTT
          V(I,J,3) = 0.
          H(I,J,3) = H(I,J,M)+AK*HT+.5*AK*AK*HTT
          RETURN
          END
```

```
      SUBROUTINE UVMISY(I,J,KB,UFY,VFY)
C     UVMISY CALCULATES U AND V ON THE FRONT IN THE Y DIRECTION
      DIMENSION DST(4)
      COMMON/A3/AX,AY
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A52/FU(30),FV(30)
      COMMON/A100/X(4),F1(4)
      COMMON/A101/F2(4)
      AI = (I-3)*AX
      DST(1) = 0.
      DO 10 IK = 2,4
      IS = KB+IK-1
      DS = SQRT((FRX(IS-1)-FRX(IS))*(FRX(IS-1)-FRX(IS))
     1 +(FRY(IS-1)-FRY(IS))*(FRY(IS-1)-FRY(IS)))
   10 DST(IK) = DST(IK-1)+DS
      DSTX = DST(2)+SQRT((FRX(KB+1)-AI)*(FRX(KB+1)-AI)
     1 +(FRY(KB+1)-FF(I))*(FRY(KB+1)-FF(I)))
      DO 20 KL=1,4
      KS = KB+KL-1
      X(KL) = DST(KL)
      F1(KL) = FU(KS)
   20 F2(KL) = FV(KS)
      CALL INTER(DSTX,4,2,UFY,VFY,DUM)
C     UFY IS U ON THE FRONT AT X COORDINATE AI
      RETURN
      END
```

```
      SUBROUTINE UVMISX(I,J,IDIR,KB,UFX,VFX,FX)
C     UVMISX CALCULATES U AND V ON THE FRONT, IN THE X DIRECTION
      DIMENSION DST(4)
      COMMON/A3/AX,AY
      COMMON/A10/EPS
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A52/FU(30),FV(30)
      COMMON/A100/X(4),F1(4)
      COMMON/A101/F2(4)
      COMMON/F1/XVP1
      COMMON/F2/MF
      AI = (I-3)*AX
      AJ = (J-1)*AY
      CALL FDIS(I,J,IDIR,KB,FX)
      KA = IFTHEN(MF .LE. 2,0,1)
      KK = KB+KA
      DST(1) = 0.
```

```
      DO 10 IK=2,4
      IS = KB+IK-1
      DS = SQRT((FRX(IS-1)-FRX(IS))*(FRX(IS-1)-FRX(IS))
     1 +(FRY(IS-1)-FRY(IS))*(FRY(IS-1)-FRY(IS)))
   10 DST(IK) = DST(IK-1)+DS
      DSTX = DST(KA+1)+SQRT((FRX(KK)-XVP1)*(FRX(KK)-XVP1)
     1 +(FRY(KK)-AJ)*(FRY(KK)-AJ))
      DO 20 KL=1,MF
      KS = KB+KL-1
      X(KL) = DST(KL)
      F1(KL) = FU(KS)
   20 F2(KL) = FV(KS)
      CALL INTER(DSTX,MF,2,UFX,VFX,DUM)
      RETURN
      END



      SUBROUTINE FDIS(I,J,IDIR,KB,FX)
C     FDIS CALCULATES THE DISTANCE TO THE FRONT IN THE X DIRECTION
      COMMON/A3/AX,AY
      COMMON/A10/EPS
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A30/ITER,ITERT
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A100/X(4),F1(4)
      COMMON/F1/XVP1
      COMMON/F2/MF
      COMMON/PR1/LP
      AI = (I-3)*AX
      AJ = (J-1)*AY
      IF (KB .GT. 0) GO TO 2
      KB = 1
      GO TO 4
    2 IF (KB .GT. NF1) KB = NF1
    4 ISPEC = 0
      L = KB+1
      CALL DER(L,DERIV)
      IF (ABS(DERIV) .GT. 1.7) GO TO 5
      LL = L+1
      CALL DER(LL,DERIV)
      IF (ABS(DERIV) .LE. 1.7) GO TO 6
    5 KB = KB+1
      MF = 2
      GO TO 10
    6 MF = 4
C     KB IS THE FRONT POINT WITH WHICH WE BEGIN OUR INTERPOLATION
```

```
 10 DO 14 IND=1,MF
    X(IND) = FRX(KB+IND-1)
 14 F1(IND) = FRY(KB+IND-1)-AJ
    ICOUNT = 0
    XVM1 = AI
    CALL INTER(XVM1,MF,1,FVM1,DUM,DUM)
    IF IDIR = 1 WE GO TO THR RIGHT
    IF IDIR = -1 WE GO TO THE LEFT
    XV = THENIF(IDIR .GE. 0.,AI+AX,AI-AX)
 60 CALL INTER(XV,MF,1,FVX,DUM,DUM)
    ICOUNT = ICOUNT+1
    IF (ICOUNT .GT. 15) GO TO 80
    XVP1 = XV-FVX*(XV-XVM1)/(FVX-FVM1)
    WE ARE USING THE METHOD OF FALSE POSITION TO SOLVE FRX(XX)-AJ=0
    IF (ABS(XV-XVP1) .LT. EPS) GO TO 80
    XVM1 = XV
    XV = XVP1
    FVM1 = FVX
    GO TO 60
 80 K = KB
 86 IF (FRX(K) .GE. XVP1) GO TO 89
    K = K+1
    IF (K .LE. KB+3) GO TO 86
    K = KB+4
 89 KK = K-1
    KK IS THE FRONT POINT SUCH THAT FRX(KK,3) IS LESS THAN XV
                 AND FRX IS AS LARGE AS POSSIBLE
    IF (MF .GT. 2) GO TO 140
    IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 600, I,J,IDIR,KB,XV,FVX
    GO TO 200
140 IF (KK .EQ. KB+1) GO TO 200
    KBT = KK-1
    IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 500, I,J,IDIR,KB,KBT,XV,FVX
    KB = KBT
    IF (KB .GT. 0) GO TO 165
    KB = 1
    GO TO 200
165 IF (KB .GT. NF1) KB = NF1
    ISPEC = ISPEC+1
    IF (ISPEC .GT. 0) MF = 2
    GO TO 10
200 FX = ABSF(AI-XVP1)
    RETURN
500 FORMAT(11H AT POINT (,I2,1H,,I2,9H) IDIR = ,I2,
   1 * KB CHANGED FROM *,I2,4H TO ,I2,6H XV = ,F11.7,7H FVX = ,F11.7)
600 FORMAT(11H AT POINT (,I2,1H,,I2,39H) WE USED LINEAR INTERPOLATION
   1 IDIR = ,I2,6H KB = ,I2,6H XV = ,F11.7,7H FVX = ,F11.7)
    END
```

```
      SUBROUTINE FRONT
      SUBROUTINE FRONT IS TO FIND THE NEW POSITION OF THE FRONT
                   AND THE VALUES OF U AND V THERE
      COMMON/A1/AL,AS,AK
      COMMON/A4/F,G
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A24/KT
      COMMON/A30/ITER,ITERT
      COMMON/A31/EPS1
      COMMON/A36/DISMIN
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A51/FHX(30),FHY(30)
      COMMON/A52/FU(30),FV(30)
      COMMON/A100/X(4),F1(4)
      DIMENSION OFU(30),OFV(30),OFRX(30),OFRY(30),OFHX(30),OFHY(30)
      FRX IS THE X COORDINATE OF THE I-TH POINT
      FHX IS H SUB X AT THE FRONT POINTS
      FU IS U AT THE FRONT POINTS
      DO 10 I=3,NF2
      IF (ITER ,GT, 1) GO TO 3
      OFU(I) = FU(I)
      OFV(I) = FV(I)
      OFRX(I) = FRX(I)
      OFRY(I) = FRY(I)
      OFHX(I) = FHX(I)
      OFHY(I) = FHY(I)
    3 FU(I) = ((1,-,25*F*F*AK*AK)*OFU(I)+F*AK*OFV(I)
     1 +,5*AK*(FHX(I)+OFHX(I))-,25*F*AK*AK*(FHY(I)+OFHY(I)-2,*G))
     2 /(1,+,25*F*F*AK*AK)
      FV(I) = (-F*AK*OFU(I)+(1,-,25*F*F*AK*AK)*OFV(I)
     1 +,25*F*AK*AK*(FHX(I)+OFHX(I))-,5*AK*(FHY(I)+OFHY(I)-2,*G))
     2 /(1,+,25*F*F*AK*AK)
      TRFX = FRX(I)
      TRFY = FRY(I)
      FRX(I) = OFRX(I)+,5*AK*(FU(I)+OFU(I))
      FRY(I) = OFRY(I)+,5*AK*(FV(I)+OFV(I))
      IF (ABS(FRX(I)-TRFX),GE,EPS1,OR,ABS(FRY(I)-TRFY),GE,EPS1) ITERT=1
   10 CONTINUE
      CALL PER2(1)
      FRS(2) = 0,
      DISMIN = 5,
      DO 100 I=3,NF3
      CALL FRONDIS(I,0,,0,,DIFDIS)
      IF (I,GT,3 ,AND, DIFDIS,LT,DISMIN) DISMIN = DIFDIS
```

```
100 FRS(I) = FRS(I-1)+DIFDIS
    CALL PER2(-1)
    RETURN
    END




    SUBROUTINE FRONDIS(I,X,Y,DIFDIS)
    SUBROUTINE FRONDIS FINDS THE DISTANCE IN TERMS OF ARCLENGTH
       BETWEEN THE FRONT POSITIONS I-1 AND (X,Y)
    COMMON/A30/ITER,ITERT
    COMMON/A50/FRX(30),FRY(30),FRS(30)
    SRP(XX) = SQRT(A*XX*XX+B*XX+C)
    DSTINT(XX) = ((2.*ALPHA*XX+BETA)*SRP(XX)
   1+ALOG(ABS(SRP(XX)+2.*ALPHA*XX+BETA)))/(4.*ALPHA)
    ITIMES = 1
    J = IABS(I)
    IF (I .LT. 0) GO TO 1
    DX = FRX(J)
    FX0 = FRX(J-1)
    FX1 = FRX(J)
    FX2 = FRX(J+1)
    FY0 = FRY(J-1)
    FY1 = FRY(J)
    FY2 = FRY(J+1)
    GO TO 5
  1 DX = X
    FX0 = FRX(J+1)
    FX1 = X
    FX2 = FRX(J+2)
    FY0 = FRY(J+1)
    FY1 = Y
    FY2 = FRY(J+2)
    IF (FX0 .NE. FX1) GO TO 4
    FX0 = FRX(J)
    FY0 = FRY(J)
  4 IF (FX1 .NE. FX2) GO TO 5
    FX2 = FRX(J+3)
    FY2 = FRY(J+3)
  5 DX01 = FX1-FX0
    DX02 = FX2-FX0
    DX12 = FX2-FX1
    TERM0 = FY0/(DX01*DX02)
    TERM1 = -FY1/(DX01*DX12)
    TERM2 = FY2/(DX02*DX12)
    ALPHA = TERM0+TERM1+TERM2
    BETA = -(TERM0*(FX1+FX2)+TERM1*(FX0+FX2)+TERM2*(FX0+FX1))
```

```
      DERIV = 2.*ALPHA*DX+BETA
      IF (ABS(DERIV) .GT. 1. .AND. ITIMES .EQ. 1) GO TO 50
      A = 4.*ALPHA*ALPHA
      B = 4.*ALPHA*BETA
      C = BETA*BETA+1.
      IF (ABS(ALPHA) .LT. .000001) GO TO 100
      IF (I .LE. 3 .AND. Y .EQ. 0.) GO TO 25
      DIFDIS = DSTINT(FX1)-DSTINT(FX0)
      IF (DIFDIS .LT. 0.) GO TO 200
      RETURN
   25 DIFDIS = DSTINT(FRX(3))-DSTINT(0.)
      RETURN
   50 DX01 = FY1-FY0
      DX02 = FY2-FY0
      DX12 = FY2-FY1
      TERM0 = FX0/(DX01*DX02)
      TERM1 = -FX1/(DX01*DX12)
      TERM2 = FX2/(DX02*DX12)
      ALPHA = TERM0+TERM1+TERM2
      BETA = -(TERM0*(FY1+FY2)+TERM1*(FY0+FY2)+TERM2*(FY0+FY1))
      A = 4.*ALPHA*ALPHA
      B = 4.*ALPHA*BETA
      C = BETA*BETA+1.
      DIFDIS = DSTINT(FY1)-DSTINT(FY0)
      IF (DIFDIS .LT. 0.) GO TO 200
      IF (DIFDIS .LT. 3.) RETURN
      ITIMES = 2
      GO TO 5
  100 IF (ITER .LE. 1) PRINT 900, I
      IF (I .LE. 3) GO TO 110
      ALPHA IS TOO SMALL TO USE REGULAR FORMULA
      DIFDIS = (FX1-FX0)*SQRT(C)
      RETURN
  110 DIFDIS = FRX(3)*SQRT(C)
      RETURN
  200 BETA = (FY1-FY0)/(FX1-FX0)
      DIFDIS = SQRT(BETA*BETA+1.)*ABS(FX1-FX0)
      RETURN
  900 FORMAT(* IN SUBROUTINE FRONDIS ALPHA WAS LESS THAN .00001 AT POINT
     1 *,I2)
      END



      SUBROUTINE FRONAM
      FRONAM IS TO FIND THE POSITION OF THE FRONT
      AT THE X COORDINATE LINES
```

```
      COMMON/A3/AX,AY
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A30/ITER,ITERT
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A75/IND(25,6)
      COMMON/A100/X(4),F1(4)
      COMMON/PR1/LP
      CALL FRONPO
      DO 100 I=3,NI1
      AI = (I-3)*AX
      NOC = 1
      NOP = 2
    5 L = IND(I,NOP)
      CALL DER(L,DERIV)
      IF (ABS(DERIV) .GE. 1.7) GO TO 50
      LL = L+1
      CALL DER(LL,DERIV)
      IF (ABS(DERIV) .GT. 1.7) GO TO 50
      KB = L-1
      DO 10 J=1,4
      X(J) = FRX(KB+J-1)
   10 F1(J) = FRY(KB+J-1)
      CALL INTER(AI,4,1,FF(I,NOC),DUM,DUM)
      GO TO 70
   50 FF(I,NOC) = FRY(L)+(FRY(L+1)-FRY(L))*(AI-FRX(L))/(FRX(L+1)-FRX(L))
      IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 900, I,L
   70 IF (NOC .GE. IND(I)) GO TO 100
      NOP = NOP+1
      NOC = NOC+1
      GO TO 5
  100 CONTINUE
      DO 200 I=3,NI1
      IF (IND(I) .LE. 1) GO TO 200
      MF = IND(I)
      MFM = MF-1
      DO 175 K=1,MFM
      MM = K
      TF = FF(I,K)
      KK = K+1
      DO 150 M=KK,MF
      IF (FF(I,M) .GT. TF) GO TO 150
      MM = M
  150 CONTINUE
      TF = FF(I,K)
      FF(I,K) = FF(I,MM)
  175 FF(I,MM) = TF
```

```
      IF (ITER .LE. 1) PRINT 950, I,(M,FF(I,M), M=1,MF)
200   CONTINUE
      RETURN
900   FORMAT(59H WE ARE USING LINEAR INTERPOLATION IN FRONAM AT COORDINA
     1TE ,I2,22H BEGINNING WITH POINT ,I2)
950   FORMAT(20H AT COORDINATE LINE ,I2,5(4H FF(,I2,4H) = ,F11.5))
      END



      SUBROUTINE POSIN(I,J,IPOS)
      COMMON/A3/AX,AY
      COMMON/A40/FF(25,6)
      COMMON/A75/IND(25,6)
      AJ = (J-1)*AY
      IF (AJ .GT. FF(I)) GO TO 1
      IPOS = 0
      RETURN
    1 IN = IND(I)
      GO TO (10,20,30,40,50), IN
   10 IPOS = 1
      RETURN
   20 IPOS = IFTHEN(AJ .LE. FF(I,2),1,0)
      RETURN
   30 IF (AJ .LE. FF(I,3)) GO TO 20
      IPOS = 1
      RETURN
   40 IF (AJ .LE. FF(I,4)) GO TO 30
      IPOS = 0
      RETURN
   50 IF (AJ .LE. FF(I,5)) GO TO 40
      IPOS = 1
      RETURN
      END



      SUBROUTINE FRONPO
      COMMON/A3/AX,AY
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A23/AKT
      COMMON/A30/ITER,ITERT
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A75/IND(25,6)
      COMMON/PP1/LP
```

```
      INDN = 0
      DO 1 ICO = 1,NI3
    1 IND(ICO) = 0
      DO 100 IFR = 1,NF3
      DO 100 ICO = 3,NI2
      AI = (ICO-3)*AX
      D1 = FRX(IFR)-AI
      D2 = FRX(IFR+1)-AI
      D3 = D1*D2
      IF (D3 .GT. 0. .OR. D1 .EQ. 0.) GO TO 100
      IF (IND(ICO) .GT. 1) INDN = INDN+1
      IND(ICO) = IND(ICO)+1
      IF (IND(ICO) .GT. 1) INDN = INDN+1
      NOP = IND(ICO)+1
      IND(ICO,NOP) = IFR
      IF (IND(ICO) .LE. 5) GO TO 100
      PRINT 900, ICO
      PRINT 960, (INO,IND(INO), INO=3,NI2)
      PRINT 970, (INO, IND(INO,2), INO=3,NI2)
      IF (INDN .LE. 0) RETURN
  100 CONTINUE
      IF (INDN .LE. 0) RETURN
      IF THE FRONT DOUBLES OVER WE PRINT THE POINT BEFORE EACH COORDINAT
      IF (ITER .LE. 1 .AND. LP .EQ. 0)
     1 PRINT 980, (INO,IND(INO,1),INO,IND(INO,2), INO=3,NI2)
      RETURN
  900 FORMAT(* AT COORDINATE *,I3,* THE FRONT CROSSED MORE THAN FIVE TIM
     1ES*)
  960 FORMAT(1X,8(5H IND(,I2,4H) = ,I3))
  970 FORMAT(1X,9(5H IND(,I2,5H,2)= ,I2))
  980 FORMAT(1X,4(5H IND(,I2,6H,1) = ,I2,5H IND(,I2,6H,2) = ,I2))
      END




      SUBROUTINE TOONER(I,J)
      TOONER FINDS THE VALUE OF U,V,H AT POINTS TOO NEAR TO THE FRONT
         FOR THE DIFFERENCE EQUATIONS TO BE USED
      COMMON/A3/AX,AY
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A15/L(24,27)
      COMMON/A24/KT
      COMMON/A30/ITER,ITERT
      COMMON/A40/FF(25,6)
      COMMON/A100/X(4),F1(4)
      COMMON/A101/F2(4)
```

```
      COMMON/A102/F3(4)
      COMMON/PP1/LP
      AI = (I-3)*AX
      AJ = (J-1)*AY
      IF (L(I,J+1) .LE. 0) GO TO 90
      IF (L(I,J) .LE. 10) GO TO 5
      IF (ITER .LE. 1) PRINT 700, I,J,L(I,J)
      GO TO 100
    5 IF (L(I,J-1)-1) 20,15,10
   10 IF (ITER .LE. 1) PRINT 950, I,J
      GO TO 100
   15 IF (ITER .LE. 1) PRINT 960, I,J
      GO TO 100
   20 IF (L(I,J+1) .EQ. 1) GO TO 25
      IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 150, H(I,J,3),I,J
      GO TO 100
   25 IF (KT .GE. 62 .AND. ITER .LE. 1 .AND. LP .EQ. 0) PRINT 500, I,J
      CALL NERFSTY(I,AJ,-1,KB)
      CALL UVMISY(I,J,KB,UFY,VFY)
      X(1) = FF(I)-AJ
      F1(1) = UFY
      F2(1) = VFY
      F3(1) = 0.
      DO 50 IN=2,3
      IK = IN-1
      X(IN) = IK*AY
      F1(IN) = U(I,J+IK,3)
      F2(IN) = V(I,J+IK,3)
   50 F3(IN) = H(I,J+IK,3)
      CALL INTER(0.,3,3,U(I,J,3),V(I,J,3),H(I,J,3))
      IF (H(I,J,3) .GT. 0.) RETURN
      IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 850, I,J
      GO TO 100
   90 IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 750, I,J
  100 IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 800
      CALL NERFST(AI,AJ,1,KB)
      CALL UVMISX(I,J,1,KB,UFX,VFX,FX)
      X(1) = FX
      F1(1) = UFX
      F2(1) = VFX
      F3(1) = 0.
      DO 150 IN=2,3
      IK = 1-IN
      X(IN) = IK*AX
      F1(IN) = U(I+IK,J,3)
      F2(IN) = V(I+IK,J,3)
  150 F3(IN) = H(I+IK,J,3)
      CALL INTER(0.,3,3,U(I,J,3),V(I,J,3),H(I,J,3))
```

```
      IF (H(I,J,3).LE.0.) CALL INTER(0.,2,3,U(I,J,3),V(I,J,3),H(I,J,3))
      RETURN
  450 FORMAT(27H H IS LESS THAN ZERO AND = ,F12.6,11H AT POINT (,
     1 I2,1H,,I2,1H))
  500 FORMAT(1X,*WE USED A REGULAR TOONER AT POINT (*,I2,1H,,I2,1H))
  700 FORMAT(11H AT POINT (,I2,1H,,I2,44H) IN TOONER L(I,J) WAS GREATER
     1THAN 9 AND = ,I2)
  750 FORMAT(11H AT POINT (,I2,1H,,I2,*)  IN TOONER L(I,J+1) WAS NOT POS
     1ITIVE*/100(1H*))
  800 FORMAT(1H+,72X,47HWE USED HORIZONTAL INTERPOLATION TO FIND TOONER)
  850 FORMAT(11H AT POINT (,I2,1H,,I2,1H))
  900 FORMAT(36H IN TOONER UPPER POINT IS MISSING AT (,I2,1H,,I2,1H),
     1 74X,12(1H*))
  950 FORMAT(11H AT POINT (,I2,1H,,I2,
     1 *)  THE POINT (I,J-1) IS NOT MISSING BUT IS TOONER*)
  960 FORMAT(11H AT POINT (,I2,1H,,I2,
     1 *)  THE POINT (I,J-1) IS NOT MISSING*)
      END



      SUBROUTINE CHGFR
      CHGFR CHECKS IF FRX IS LESS THAN 0 OR GREATER THAN 20
      COMMON/A20/TLEN
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A23/AKT
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A52/FU(30),FV(30)
   10 IF (FRX(NF2) .LT. TLEN) GO TO 40
      K = 3
      TRFX = FRX(NF2)-TLEN
      TRFY = FRY(NF2)
      TRFU = FU(NF2)
      TRFV = FV(NF2)
   20 IF (FRX(K) .GE. TRFX) GO TO 25
      K = K+1
      GO TO 20
   25 KK = NF2-K
      DO 30 J=1,KK
      JF = NF2-J
      FRX(JF+1) = FRX(JF)
      FRY(JF+1) = FRY(JF)
      FU(JF+1) = FU(JF)
   30 FV(JF+1) = FV(JF)
      FRX(K) = TRFX
      FRY(K) = TRFY
      FU(K) = TRFU
```

```
      FV(K) = TRFV
      PRINT 200, AKT,K
      GO TO 10
 40   IF (FRX(3) .GE. 0.) GO TO 100
      TRFX = FRX(3)+TLEN
      TRFY = FRY(3)
      TRFU = FU(3)
      TRFV = FV(3)
      K = NF2
 45   IF (FRX(K) .LE. TRFX) GO TO 50
      K = K-1
      GO TO 45
 50   KK = K-1
      DO 60 J=3,KK
      FRX(J) = FRX(J+1)
      FRY(J) = FRY(J+1)
      FU(J) = FU(J+1)
 60   FV(J) = FV(J+1)
      FRX(K) = TRFX
      FRY(K) = TRFY
      FU(K) = TRFU
      FV(K) = TRFV
      PRINT 201, AKT,K
      GO TO 40
100   CALL PER2(1)
      FRS(2) = 0.
      DO 150 I=3,NF3
      CALL FRONDIS(I,0.,0.,DIFDIS)
150   FRS(I) = FRS(I-1)+DIFDIS
      CALL PER2(-1)
      RETURN
200   FORMAT(29H FRX EXCEEDED TWENTY AT TIME ,F8.2,
     1 18H AND BECAME POINT ,I3)
201   FORMAT(32H FRX WAS LESS THAN ZERO AT TIME ,F8.2,
     1 18H AND BECAME POINT ,I3)
      END



      SUBROUTINE FRONFH
C     SUBROUTINE FRONFH IS TO FIND H SUB X AND H SUB Y AT THE FRONT
      DIMENSION HM(2),DIS(2)
      COMMON/A3/AX,AY
      COMMON/A6/H(24,27,3)
      COMMON/A15/L(24,27)
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
```

```
      COMMON/A24/KT
      COMMON/A30/ITER,ITER1
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A51/FHX(30),FHY(30)
      COMMON/A100/X(4),F1(4)
      COMMON/PR1/LP
      KP = 500
C     WE ARE TRYING TO FIND THE DERIVATIVES OF H AT THE FRONT
C     WE FIRST FIND THE TANGENT TO THE FRONT AT POINT I
C     WE FIND THIS TANGENT BY USING CUBIC INTERPOLATION
      DO 300 I=3,NF2
      CALL DER(I,DERIV)
      IF (ABS(DERIV) .LT. .000001) DERIV = .000001
      SLOPE = -1./DERIV
C     SLOPE IS THE SLOPE OF THE NORMAL TO THE FRONT AT POINT I
      XX = SQRT(1.+SLOPE*SLOPE)
      JPT = FRY(I)/AY+1.
      IF (ABS(SLOPE) .LT. 1.) GO TO 100
      FJP = JPT*AY
      DST = ABS(XX*(FJP-FRY(I))/SLOPE)
      IF (KP .EQ. 0) PRINT 920, I,SLOPE,DST
      KH = IFTHEN(DST .GE. .25*AX,0,1)
      DO 99 JIN=1,2
      JPR = JPT+KH+JIN-1
      FJP = JPR*AY
      JPZ = JPR+1
      XS = FRX(I)+(FJP-FRY(I))/SLOPE
      DIS(JIN) = ABS(XX*(FJP-FRY(I))/SLOPE)
C     XS IS THE X COORDINATE OF THE NORMAL AT Y COORDINATE JPZ
C     DIS IS THE DISTANCE BETWWEN FRONT POINT I AND THE POINT (XS,JPZ)
      IPO = XS/AX+3.
      IP1 = IPO+1
      IF (IPO .GT. 0) GO TO 5
      PRINT 985, I,IPO,JPZ,SLOPE,FRX(I),FRY(I)
      CALL MYEXIT(5)
    5 IF (IPO .LT. NI2) GO TO 10
      NUM = IFTHEN(IPO .EQ. NF2,3,2)
      KBO = 1
      GO TO 40
   10 IF (L(IPO,JPZ) .GT. 0) GO TO 15
      KST = I-2
      CALL FDIS(IP1,JPZ,-1,KST,FX)
C     FX IS THE DISTANCE BETWEEN IP1 AND THE FRONT CURVE CROSSING JPZ
      X(1) = IP1-FX/AX
      F1(1) = 0.
      DO 12 IK=2,3
      KN = IPO+IK-1
```

```
      X(IK) = KN
   12 F1(IK) = H(KN,JPZ,3)
      NUM = 3
      IF (KP .EQ. 0) PRINT 940, IPO,JPZ,FJP,XS,JIN,DIS(JIN)
      GO TO 50
   15 IF (L(IPO-1,JPZ) .GT. 0) GO TO 20
      FIP = (IPO-3)*AX
      CALL NEREST(FIP,FJP,-1,KST)
      CALL FDIS(IPO,JPZ,-1,KST,FX)
      X(1) = IPO-FX/AX
      F1(1) = 0.
      DO 17 IK=2,4
      KN = IPO+IK-2
      X(IK) = KN
   17 F1(IK) = H(KN,JPZ,3)
      NUM = 4
      IF (KP .GT. 0) GO TO 50
      IPOP = IPO-1
      PRINT 940, IPOP,JPZ,FJP,XS,JIN,DIS(JIN)
      GO TO 50
   20 IF (L(IP1,JPZ) .GT. 0) GO TO 25
      KST = I-1
      CALL FDIS(IPO,JPZ,1,KST,FX)
      DO 22 IK=1,2
      KN = IPO+IK-2
      X(IK) = KN
   22 F1(IK) = H(KN,JPZ,3)
      X(3) = X(2)+FX/AX
      F1(3) = 0.
      NUM = 3
      IF (KP .EQ. 0) PRINT 940, IP1,JPZ,FJP,XS,JIN,DIS(JIN)
      GO TO 50
   25 IF (L(IPO+2,JPZ) .GT. 0) GO TO 30
      FIP = (IP1-3)*AX
      CALL NEREST(FIP,FJP,1,KST)
      CALL FDIS(IP1,JPZ,1,KST,FX)
      DO 29 IK=1,3
      KN = IPO+IK-2
      X(IK) = KN
   29 F1(IK) = H(KN,JPZ,3)
      X(4) = X(3)+FX/AX
      F1(4) = 0.
      NUM = 4
      IF (KP .GT. 0) GO TO 50
      IPOP = IPO+2
      PRINT 940, IPOP,JPZ,FJP,XS,JIN,DIS(JIN)
      GO TO 50
   30 KBO = 1
```

```
      NUM = 4
      IF (KP .EQ. 0) PRINT 941, FJP,XS,JIN,DIS(JIN)
   40 DO 41 IK=1,NUM
      KN = IPO-KBO+IK-1
      X(IK) = KN
   41 F1(IK) = H(KN,JPZ,3)
   50 XA = XS/AX+3.
      IF (KP .GT. 0) GO TO 99
      PRINT 960, XA,(IK,X(IK), IK=1,NUM)
      PRINT 961, (IK,F1(IK), IK=1,NUM)
   99 CALL INTER(XA,NUM,1,HH(JIN),DUM,DUM)
      DX = XS-FRX(I)
      DY = FJP-FRY(I)
      GO TO 200
  100 IPT = FRX(I)/AX
      FIP = IPT*AX
      IS = -1
      IF (FRY(I+1) .GE. FRY(I)) GO TO 104
      IPT = IPT+1
      FIP = FIP+1
      IS = 1
  104 DST = ABS(XX*(FIP-FRX(I)))
      IF (ITER .LE. 1 .AND. LP .EQ. 0) PRINT 920, I,SLOPE,DST
      KH = IFTHEN(DST .LT. .25*AX,IS,0)
      DO 199 JIN=1,2
      IPR = IPT+KH+(JIN-1)*IS
      FIP = IPR*AY
      IPZ = IPR+3
      YS = FRY(I)+SLOPE*(FIP-FRX(I))
      DIS(JIN) = ABS(XX*(FIP-FRX(I)))
      JPO = YS/AY+1
      IF (L(IPZ,JPO) .GT. 0) GO TO 122
      KBO = 1
      NUM = 3
      IF (KP .EQ. 0) PRINT 950, IPZ,JPO,FIP,YS,JIN,DIS(JIN)
      GO TO 127
  122 IF (L(IPZ,JPO-1) .GT. 0) GO TO 130
      KBO = 2
      NUM = 4
      IF (KP .GT. 0) GO TO 127
      JPOP = JPO-1
      PRINT 950, IPZ,JPOP,FIP,YS,JIN,DIS(JIN)
  127 X(1) = FF(IPZ)+1
      F1(1) = 0.
      DO 128 IK=2,NUM
      KN = JPO+IK-KBO
      X(IK) = KN
  128 F1(IK) = H(IPZ,KN,3)
```

```
      GO TO 150
  130 IF (L(IPZ,JPO+1) .GT. 0) GO TO 133
      KBO = 2
      NUM = 2
      JPOP = JPO+1
      PRINT 950, IPZ,JPOP,FIP,YS,JIN,DIS(JIN)
      GO TO 137
  133 IF (L(IPZ,JPO+2) .GT. 0) GO TO 140
      KBO = 2
      NUM = 3
      IF (KP .GT. 0) GO TO 137
      JPOP = JPO+2
      PRINT 950, IPZ,JPOP,FIP,YS,JIN,DIS(JIN)
  137 DO 139 IK=1,NUM
      KN = JPO+IK-KBO
      X(IK) = KN
  139 F1(IK) = H(IPZ,KN,3)
      NUM = NUM+1
      X(NUM) = FF(IPZ)+1.
      F1(NUM) = 0.
      GO TO 150
  140 DO 145 IK=1,4
      KN = JPO+IK-2
      X(IK) = KN
  145 F1(IK) = H(IPZ,KN,3)
      NUM = 4
      IF (KT .GT. KP) PRINT 951, FIP,YS,JIN,DIS(JIN)
  150 XA = YS/AY+1
      IF (KP .GT. 0) GO TO 199
      PRINT 960, XA, (IK,X(IK),IK=1,NUM)
      PRINT 961, (IK,F1(IK), IK=1,NUM)
  199 CALL INTER(XA,NUM,1,HM(JIN),DUM,DUM)
      DX = FIP-FRX(I)
      DY = YS-FRY(I)
C     HM IS THE VALUE OF H AT POINTS ABOVE THE FRONT
  200 H1 = DIS(1)
      H2 = DIS(2)
      DH = H2-H1
      FN = H2*HM(1)/(H1*DH)-H1*HM(2)/(H2*DH)
C     FN IS H SUB N AT THE FRONT
      FX = SIGN(1.,DX)*FN/XX
      FY = SIGN(1.,DY)*FN*ABS(SLOPE)/XX
      IF (KP .EQ. 0) PRINT 970, DX,DY,HM(1),HM(2),XX,FN
      FHX(I) = FX
  300 FHY(I) = FY
C     FHX IS THE X DERIVATIVE OF H AT THE FRONT
C     FHY IS THE Y DERIVATIVE OF H AT THE FRONT
      RETURN
```

```
920 FORMAT(40X,9HAT POINT ,I2,10H  SLOPE = ,F12.7,7H DST = ,F12.7)
940 FORMAT(8H POINT (,I2,1H,,I2,21H) WAS MISSING  FJP = ,F12.6,
   1 6H XS = ,F12.6,5H DIS(,I1,4H) = ,F12.6)
941 FORMAT(23H NO POINTS WERE MISSING,4X,7H FJP = ,F12.6,
   1 6H XS = ,F12.6,5H DIS(,I1,4H) = ,F12.6)
950 FORMAT(8H POINT (,I2,1H,,I2,21H) WAS MISSING  FIP = ,F12.6,
   1 6H YS = ,F12.6,5H DIS(,I1,4H) = ,F12.6)
951 FORMAT(23H NO POINTS WERE MISSING,4X,7H FIP = ,F12.6,
   1 6H YS = ,F12.6,5H DIS(,I1,4H) = ,F12.6)
960 FORMAT(6H XA = ,F12.6,4(4H  X(,I2,4H) = ,F12.6))
961 FORMAT(18X,4(4H F1(,I2,4H) = ,F12.6))
962 FORMAT(6H FN = ,F14.8,6H XX = F12.6)
970 FORMAT(6H DX = ,F10.6,6H DY = ,F10.6,9H HM(1) = ,F10.6,
   1 9H HM(2) = ,F10.6,6H XX = ,F10.6,6X,6H FN = ,F12.8)
980 FORMAT(1X,*WE ATTEMPTED TO GO BEYOND THE END OF THE REGION    WITH
   1 IPO = *,I3)
985 FORMAT(33H IPO WAS LESS THAN ZERO AT POINT ,I2,12H WITH IPO = ,I2,
   1 7H JPZ = ,I2,9H SLOPE = ,F12.6,7H FRX = ,F12.6,7H FRY = ,F12.6/
   2 100(1H*))
    END



    SUBROUTINE DER(L,DERIV)
    COMMON/A24/KT
    COMMON/A50/FRX(30),FRY(30),FRS(30)
    DIMENSION D(2)
    X0 = FRS(L-1)
    X1 = FRS(L)
    X2 = FRS(L+1)
    TERM0 = (X1-X2)/((X0-X1)*(X0-X2))
    TERM1 = (2.*X1-X0-X2)/((X1-X0)*(X1-X2))
    TERM2 = (X1-X0)/((X2-X0)*(X2-X1))
    DO 3 ISL=1,2
    IF (ISL .GT. 1) GO TO 2
    FX0 = FRY(L-1)
    FX1 = FRY(L)
    FX2 = FRY(L+1)
    GO TO 3
  2 FX0 = FRX(L-1)
    FX1 = FRX(L)
    FX2 = FRX(L+1)
  3 D(ISL) = FX0*TERM0+FX1*TERM1+FX2*TERM2
    DERIV = D(1)/D(2)
    RETURN
    END
```

```
      SUBROUTINE NEREST(AI,AJ,IDIR,KB)
C     NEREST FINDS THE NEAREST FRONT POINT TO THE COLUMN AI
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A50/FRX(30),FRY(30),FRS(30)
C     IF IDIR IS POSITIVE THEN FRONT CURVE IS TO THE RIGHT OF MESH POINT
C     IF IDIR IS NEGATIVE THEN FRONT CURVE IS TO THE LEFT OF MESH POINT
      MM = 3
C     WE ARE FINDING THE NEAREST POINT ON THE FRONT TO (AI,AJ)
      SM = SQRT((FRX(3)-AI)*(FRX(3)-AI)+(FRY(3)-AJ)*(FRY(3)-AJ))
      DO 5 M=4,NF2
      ASM = SQRT((FRX(M)-AI)*(FRX(M)-AI)+(FRY(M)-AJ)*(FRY(M)-AJ))
      IF (ASM .GE. SM) GO TO 5
      SM = ASM
      MM = M
    5 CONTINUE
      IF (FRY(MM) .LT. AJ) GO TO 22
      KB = IFTHEN(IDIR .LE. 0,MM-1,MM-2)
      RETURN
C     IF IDIR IS 1 WE GO TO THE RIGHT, OTHERWISE TO THE LEFT
   22 IF (IDIR .LE. 0) GO TO 30
      K = MM
   24 K = K+1
      IF (K .GT. NF2) GO TO 60
      IF (FRY(K) .LT. AJ) GO TO 24
      KB = K-2
      RETURN
   30 K = MM+1
   31 K = K-1
      IF (K .LT. 2) GO TO 60
      IF (FRY(K) .LT. AJ) GO TO 31
      KB = K-1
      RETURN
   60 KB = IFTHEN(FRX(MM) .LE. AI,MM-1,MM-2)
C     IF FRX(MM) IS LESS THAN AI, WE USE THE POINTS  MM-1,MM,MM+1,MM+2
C     IF FRX(MM) IS GREATER THAN AI WE USE THE POINTS MM-2,MM-1,MM,MM+1
C     KB IS THE FIRST FRONT POINT USED IN THE INTERPOLATION
      RETURN
      END


      SUBROUTINE NERESTY(I,AJ,IDIR,KB)
      COMMON/A40/FF(25,6)
      COMMON/A75/IND(25,6)
C     IF IDIR IS POSITIVE THEN FRONT CURVE IS ABOVE THE MESH POINT
C     IF IDIR IS NEGATIVE THEN FRONT CURVE IS BELOW THE MESH POINT
      IF (IND(I) .GT. 1) GO TO 10
```

```
      NOC = 1
      GO TO 100
   10 IF (IDIR ,GT, 0) GO TO 50
      NOC = IND(I)+1
   15 NOC = NOC-1
      IF (AJ ,LT, FF(I,NOC)) GO TO 15
      GO TO 100
   50 NOC = 1
   55 NOC = NOC+1
      IF (AJ ,GE, FF(I,NOC)) GO TO 55
  100 KB = IND(I,NOC+1)-1
      RETURN
      END



      SUBROUTINE PERIOD(M)
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      DO 1 J=1,NJ
      U(1,J,M) = U(NI,J,M)
      V(1,J,M) = V(NI,J,M)
      H(1,J,M) = H(NI,J,M)
      U(2,J,M) = U(NI1,J,M)
      V(2,J,M) = V(NI1,J,M)
      H(2,J,M) = H(NI1,J,M)
      U(NI2,J,M) = U(3,J,M)
      V(NI2,J,M) = V(3,J,M)
      H(NI2,J,M) = H(3,J,M)
      U(NI3,J,M) = U(4,J,M)
      V(NI3,J,M) = V(4,J,M)
    1 H(NI3,J,M) = H(4,J,M)
      RETURN
      END



      SUBROUTINE PER2(IBEGIN)
      COMMON/A20/TLEN
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A52/FU(30),FV(30)
C     PER2 EVALUATES THE FRONT VALUES IN ACCORDANCE WITH PERIODICITY
C     IF IBEGIN IS 0 WE DO ALL OF PER2
C     IF IBEGIN IS NEGATIVE WE DO SECTION 1 ONLY
```

```
C     IF IBEGIN IS POSITIVE WE DO SECTION 2 ONLY
      IF (IBEGIN .GT. 0) GO TO 10
C                                 SECTION 1
      FU(1) = FU(NF1)
      FU(2) = FU(NF2)
      FU(NF3) = FU(3)
      FU(NF4) = FU(4)
      FV(1) = FV(NF1)
      FV(2) = FV(NF2)
      FV(NF3) = FV(3)
      FV(NF4) = FV(4)
      FRS(1) = FRS(3)+FRS(NF1)-FRS(NF3)
      FRS(2) = FRS(3)+FRS(NF2)-FRS(NF3)
      FRS(NF4) = FRS(NF3)+FRS(4)-FRS(3)
      FRX(1) = FRX(NF1)-TLEN
      FRY(1) = FRY(NF1)
      IF (IBEGIN .LT. 0) RETURN
C                                 SECTION 2
   10 FRX(2) = FRX(NF2)-TLEN
      FRY(2) = FRY(NF2)
      FRX(NF3) = FRX(3)+TLEN
      FRY(NF3) = FRY(3)
      FRX(NF4) = FRX(4)+TLEN
      FRY(NF4) = FRY(4)
      RETURN
      END



      SUBROUTINE INTER(XX,NUMP,NUMBER,AIN1,AIN2,AIN3)
C     INTER PERFORMS LAGRANGE INTERPOLATION
      COMMON/A100/X(4),F1(4)
      COMMON/A101/F2(4)
      COMMON/A102/F3(4)
C     NUMP IS THE NUMBER OF INTERPOLATING POINTS USED
C     NUMBER IS THE NUMBER OF FUNCTIONS DESIRED
      REAL NUM
      AIN1 = 0.
      AIN2 = 0.
      AIN3 = 0.
      DO 10 KL=1,NUMP
      NUM = 1.0
      DEN = 1.0
      DO 4 JL=1,NUMP
      IF (KL .EQ. JL) GO TO 4
      NUM = NUM*(XX-X(JL))
      DEN = DEN*(X(KL)-X(JL))
```

```
    4 CONTINUE
      IF (ABS(DEN) .GE. .0000001) GO TO 7
      PRINT 950,DEN,NUM,KL
      GO TO 60
    7 DNUM = NUM/DEN
      AIN1 = AIN1+F1(KL)*DNUM
      IF (NUMBER-2) 10,8,9
    8 AIN2 = AIN2+F2(KL)*DNUM
      GO TO 10
    9 AIN2 = AIN2+F2(KL)*DNUM
      AIN3 = AIN3+F3(KL)*DNUM
   10 CONTINUE
      RETURN
   60 PRINT 960, (IK,X(IK), IK=1,NUMP)
      PRINT 961, (IK,F1(IK), IK=1,NUMF)
      PRINT 962, (IK,F2(IK), IK=1,NUMF)
      PRINT 998
      CALL EXIT
  950 FORMAT(8H DEN =  ,F12.5,10H    NUM =  ,F12.5,7H  TIME ,I2)
  960 FORMAT(1X,4(4H  X(,I2,4H) = ,F14.6))
  961 FORMAT(1X,4(4H F1(,I2,4H) = ,F14.8))
  962 FORMAT(1X,4(4H F2(,I2,4H) = ,F14.8))
  998 FORMAT(50(1H*))
      END



      SUBROUTINE RELABLE
      SUBROUTINE RELABLE REDISTRIBUTES THE POINTS ALONG THE FRONT
      COMMON/A20/TLEN
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A52/FU(30),FV(30)
      COMMON/A100/X(4),F1(4)
      COMMON/A101/F2(4)
      COMMON/PR1/LP
      DIMENSION KB(30)
      DIMENSION TFRX(30),TFRY(30),TFRS(30),OFRS(30)
      DIMENSION TFU(30),TFV(30)
      EQUIVALENCE (TFU(1),TFRX(1)),(TFV(1),TFRY(1))
      DELS = (FRS(NF3)-FRS(3))/NF
      PRINT 500, DELS
      TFRX(3) = 0.
      TFRS(3) = 0.
      DISTANCE IS MEASURED FROM THE POINT (0,Y(3))
      DO 1 K=1,3
      X(K) = FRS(K+1)
```

```
 1 F1(K) = FRY(K+1)
   CALL INTER(0.,3,1,TFRY(3),DUM,DUM)
C  WE HAVE JUST FOUND TFRY(3)
   DO 2 I=4,NF2
 2 TFRS(I) = TFRS(I-1)+DELS
   KB(3) = 1
   DO 5 I=4,NF2
   K = I-4
 3 K = K+1
   IF (K .GE. NF3) GO TO 5
   IF (FRS(K) .LT. TFRS(I)) GO TO 3
 5 KB(I) = K-2
   DO 10 I=4,NF2
   DO 9 K=1,4
   L = KB(I)+K-1
   X(K) = FRS(L)
   F1(K) = FRX(L)
 9 F2(K) = FRY(L)
10 CALL INTER(TFRS(I),4,2,TFRX(I),TFRY(I),DUM)
C  WE HAVE JUST FOUND TFRX AND TFRY
   DO 15 I=1,NF4
15 OFRS(I) = FRS(I)
   DO 20 I=3,NF2
   FRS(I) = TFRS(I)
   FRX(I) = TFRX(I)
20 FRY(I) = TFRY(I)
   FRX(NF3) = TLEN
   FRX(NF4) = FRX(4)+TLEN
   FRY(NF3) = FRY(3)
   FRY(NF4) = FRY(4)
   CALL FRONDIS(NF3,0.,0.,DIFDIS)
   FRS(NF3) = FRS(NF2)+DIFDIS
   FRS(1) = FRS(3)+FRS(NF1)-FRS(NF3)
   FRS(2) = FRS(3)+FRS(NF2)-FRS(NF3)
   FRS(NF4) = FRS(NF3)+FRS(4)-FRS(3)
C  WE HAVE JUST FOUND FRS AT THE CORNERS
   DO 30 I=3,NF2
   DO 25 K=1,4
   L = KB(I)+K-1
   X(K) = OFRS(L)
   F1(K) = FU(L)
25 F2(K) = FV(L)
30 CALL INTER(FRS(I),4,2,TFU(I),TFV(I),DUM)
C  WE HAVE JUST FOUND FU AND FV
   DO 40 I=3,NF2
   FU(I) = TFU(I)
40 FV(I) = TFV(I)
   CALL PFR2(0)
```

```
      RETURN
  500 FORMAT(1X*SUBROUTINE RELABLE        DELS = *,F12.7)
      END



      SUBROUTINE MYPRNT1(IWHERE)
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A40/FF(25,6)
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      COMMON/A51/FHX(30),FHY(30)
      COMMON/A52/FU(30),FV(30)
      IF IWHERE IS 1 WE PRINT FRX AND FU ONLY
      IF IWHERE IS 2 WE PRINT FRX,FU AND FF
      IF IWHERE IS 3 WE PRINT EVERYTHING
      PRINT 920, (I,FRX(I),FRY(I),FRS(I),FU(I),FV(I), I=1,NF4)
      PRINT 997
      IF (IWHERE .LE. 1) RETURN
      PRINT 930, (I,FF(I), I=3,NI1)
      PRINT 997
      IF (IWHERE .LE. 2) RETURN
      PRINT 950, (I,FHX(I),I,FHY(I), I=3,NF2)
      RETURN
  920 FORMAT(16H AT FRONT POINT ,I3,7H   X = ,F12.6,7H   Y = ,F12.6,
     1 7H   S = ,F12.6,7H   U = ,E12.4,7H   V = ,E12.4)
  930 FORMAT(1X,5(5H  FF(,I2,4H) = ,E12.4))
  950 FORMAT(1X,2(5H FHX(,I2,4H) = ,F12.6,7H   FHY(,I2,4H) = ,F12.6))
  997 FORMAT(1H )
      END



      SUBROUTINE MYPRNT2(M)
      COMMON/A5/U(24,27,3),V(24,27,3)
      COMMON/A6/H(24,27,3)
      COMMON/A21/NI,NJ,NI1,NI2,NI3
      PRINT 460
      PRINT 480, (J, (H(I,J,M), I=3,NI1), J=1,NJ)
      PRINT 461
      PRINT 480, (J, (U(I,J,M), I=3,NI1), J=1,NJ)
      PRINT 462
      PRINT 480, (J, (V(I,J,M), I=3,NI1), J=1,NJ)
      RETURN
  460 FORMAT(//60X,1HH/3H I=,6X,1H3,11X,1H4,11X,1H5,11X,1H6,11X,1H7,11X,
     1 1H8,11X,1H9,11X,2H10,10X,2H11,10X,2H12//)
```

```
461 FORMAT(//60X,1HU/3H I=,6X,1H3,11X,1H4,11X,1H5,11X,1H6,11X,1H7,11X,
   1 1H8,11X,1H9,11X,2H10,10X,2H11,10X,2H12//)
462 FORMAT(//60X,1HV/3H I=,6X,1H3,11X,1H4,11X,1H5,11X,1H6,11X,1H7,11X,
   1 1H8,11X,1H9,11X,2H10,10X,2H11,10X,2H12//)
480 FORMAT(1X,I2,10(E11.4,1X)/3X,10(E11.4,1X))
997 FORMAT(1H )
    END



    SUBROUTINE MYPLOT
    COMMON/A20/TLEN
    COMMON/A21/NI,NJ,NI1,NI2,NI3
    COMMON/A22/NF,NF1,NF2,NF3,NF4
    COMMON/A23/AKT
    COMMON/A50/FRX(30),FRY(30),FRS(30)
    COMMON/PL/SIZEX,SIZEY,TLENY
    YMIN IS THE MINIMUM OF Y TO BE PLOTTED
    SIZE IS THE LENGTH OF THE AXIS IN INCHES
    TLEN IS THE NUMBER OF COORDINATE LINES ON THE AXES
    SFX IS THE SCALE FACTOR FOR THE X AXIS
    DISX IS THE LENGTH BETWEEN PLOTS IN INCHES
    DISX = SIZEX+2.5
    YMIN = 6.
    SFX = TLEN/SIZEX
    SFY = (TLENY-YMIN)/SIZEY
    LF = NF+1
    DIVX = 25.4
    DIVY = 25.4
    DIV = 10*RANGE/(LENGTH*NUMBER OF COORDINATE LINES PER TIC)
    CALL AXIS(0.,0.,1HX,-1,SIZEX,0.,0.,SFX,DIVX)
    CALL AXIS(0.,0.,1HY,1,SIZEY,90.,YMIN,SFY,DIVY)
    CALL LINE(FRX(3),FRY(3),LF,1,1,11,0.,SFX,YMIN,SFY)
    CALL SYMBOL(.75,5.0,.28,5HHOURS,0.,5)
    AHR = AKT/60.
    CALL NUMBER(.75,5.5,.28,AHR,0.,2)
    CALL PLOT(DISX,0.,-3)
    PRINT 100, AKT
    CALL CONT(NI,NJ)
    RETURN
100 FORMAT(9H AT TIME ,F12.7,* WE COMPLETED A PLOT*)
    END



    SUBROUTINE MYEXIT(N)
```

```
      COMMON/A25/IPRINT,IPLOT
      COMMON/PR1/LP
      IF (LP .EQ. 0) PRINT 100
      IF (LP .EQ. 0) CALL MYPRNT1(3)
      IF (IPLOT .LE. 0) GO TO 50
      CALL PLOT(5.,0.,-3)
      CALL PLOT(0.,0.,999)
   50 PRINT 200, N
      STOP
  100 FORMAT(19H WE ARE NOW EXITING)
  200 FORMAT(6H STOP ,I2)
      END
```

```
      SUBROUTINE CONT(NI,NJ)
      COMMON/A6/H(24,27,3)
      COMMON/A22/NF,NF1,NF2,NF3,NF4
      COMMON/A50/FRX(30),FRY(30),FRS(30)
      DIMENSION HT(27,21),CL(5)
      DIMENSION ITITLE(2),LABELX(2),LABELY(2)
      EQUIVALENCE (HT(1,1),H(1,1,2))
      EXTERNAL FX
      DO 1 I=1,NI
      DO 1 J=1,NJ
    1 HT(J,I) = H(I+2,J,3)
      KDIM = 27
      NNI = NI
      MNJ = NJ
      XA = 0.
      XB = NI-1
      YA = 0.
      YB = NJ-1
      XG = 5.
      YG = 6.5
      NCL = 4
      DO 2 I=1,NCL
    2 CL(I) = .15552*I
      ITITLE(1) = 10HCONTOURS
      ITITLE(2) = 0
      LABELX(1) = 6HX AXIS
      LABELX(2) = 0
      LABELY(1) = 6HY AXIS
      LABELY(2) = 0
      KP = 0
      CALL CONTOUR(HT,KDIM,MNJ,NNI,MNJ,NNI,XA,XB,YA,YB,XG,YG,NCL,CL,
     1 ITITLE,LABELX,LABELY,FX,FX,KP)
```

```
      XSF = (NI-1)/XG
      YSF = (NJ-1)/YG
      LF = NF+1
      CALL LINE(FRX(3),FRY(3),LF,1,0,1,0,,XSF,0,,YSF)
      CALL PLOT(0,,-11,,-3)
      CALL PLOT(10,5,1,,-3)
      RETURN
      END



      FUNCTION FX(X)
      FX = X
      RETURN
      END



      SUBROUTINE CONTOUR(Z,KDIM,
     1 M,N,MM,NN,XA,XB,YA,YB,XG,YG,NCL,CL,ITITLE,LABELX,LABELY,FX,FY,KP)
      DATA TWOPI/6,2831853071795B/
      COMMON/POLARC/RS,R0,THS,TH0
      COMMON/INDICES/MROW,NCOL,MMROW,NNCOL,KPOL
      COMMON/XYBNDS/XMIN,XMAX,YMIN,YMAX,XSIZE,YSIZE,
     1HX,HY,XS,XSS,YS,YSS,FXA,FYA
      COMMON/CLEVELS/NLVLS,NLV,CLEVEL(50)
      COMMON/CAVIN/IDIM,DUM(4035)
      DIMENSION Z(1)
      DIMENSION CL(1)
      Z(I,J) IS THE ORDINATE AT POINT X(J), Y(I)
      I VARIES BETWEEN 1 AND M
      J VARIES BETWEEN 1 AND N
      Z HAS DIMENSION (KDIM,,)
      MXN IS THE SIZE OF THE CALCULATED X-Y GRID
      MMXNN IS THE SIZE OF THE EXPANDED(BY INTERPOLATION) X-Y GRID
      XA,XB,YA,YB ARE THE MINIMUM AND MAXIMUM VALUES
           OF X AND Y,
      XG IS THE WIDTH OF THE GRAPH  IN INCHES,
      YG IS THE HEIGHT OF THE GRAPH IN INCHES,
      NCL IS THE NO, OF CONTOUR LEVELS
      CL(I) ARE THE CONTOUR LEVELS
      ITITLE CONTAINS THE PLOT TITLES     IT SHOULD END WITH ZERO WORD
      LABELX CONTAINS THE LABELLING ALONG THE X AXIS
      LABELY CONTAINS THE LABELLING ALONG THE Y AXIS
      THE X(I) ARE ASSUMED TO BE EQUALLY SPACED, AND
      LIKEWISE, THE Y(I),
```

```
C     FX IS THE FUNCTION TO BE PLOTTED ALONG THE X-AXIS,
C     FY IS THE FUNCTION TO BE PLOTTED ALONG THE Y-AXIS,
C     IF KP = 0 CARTESIAN COORDINATES ARE USED
C     OTHERWISE POLAR COORDINATES ARE USED,
      IDIM=KDIM
      MROW=M
      NCOL = N
      MMROW = MM
      NNCOL = NN
      KPOL=0
      IF (KP .NE. 0) GO TO 1
      XMIN=XA
      XMAX = XB
      YMIN = YA
      YMAX = YB
      GO TO 2
    1 XMIN=-YB
      XMAX=YB
      YMIN=XMIN
      YMAX=XMAX
      THO=XA
      RO=YA
      THS=(XB-XA)/FLOAT(NNCOL-1)
      RS=(YB-YA)/FLOAT(MMROW-1)
      KPOL=1
    2 CONTINUE
      XSIZE=XG
      YSIZE = YG
      NLVLS = IABS(NCL)
      CALL PLOT(0.,-.5*(11.-YSIZE),3)
      IF (NCL .GE. 0) GO TO 12
      CLEVEL = Z
      HX = Z
      L = 0
      DO 8 I=1,NCOL
      DO 7 J=1,MROW
      L = L+1
      IF (Z(L) .GE. CLEVEL) GO TO 5
      CLEVEL = Z(L)
    5 IF (Z(L) .LE. HX) GO TO 7
      HX = Z(L)
    7 CONTINUE
    8 L=L-M+IDIM
      HX = (HX-CLEVEL)/FLOAT(NLVLS-1)
      DO 10 I=2,NLVLS
      CLEVEL(I)=CLEVEL(I-1)+HX
   10 CONTINUE
      GO TO 20
```

```
 12 DO 15 I=1,NLVLS
 15 CLEVEL(I) = CL(I)
 20 HX=(XMAX-XMIN)/FLOAT(NCOL-1)
    HY=(YMAX-YMIN)/FLOAT(MROW-1)
    XS=(XMAX-XMIN)/FLOAT(NNCOL-1)
    YS=(YMAX-YMIN)/FLOAT(MMROW-1)
    FXA=FX(XMIN)
    FYA = FY(YMIN)
    XSS=XG/(FX(XMAX)-FXA)
    YSS=YG/(FY(YMAX)-FYA)
    CALL INTERP(Z,IDIM)
    DO 30 NLV=1,NLVLS
 30 CALL SCAN(Z,FX,FY)
    CALL LABEL(ITITLE,LABELX,LABELY,FX,FY)
    IF (KPOL .EQ. 0) GO TO 200
    XCENT=XSIZE*.5
    YCENT=YSIZE*.5
    RZERO=XCENT*RO/YP
    RMAX=XCENT
    THETA1=XR
    IF (THETA1 .GT. TWOPI) THETA1= TWOPI+AMOD(THETA,TWOPI)
    IF (RZERO .GT. 1.) CALL ARC(XCENT,YCENT,RZERO,THO,THETA1,1)
    CALL ARC(XCENT,YCENT,RMAX,THO,THETA1,1)
    THH = THO
    K = 2
    IF(ABS(THETA1-(TWOPI+THO)).LT..001) K = 1
    DO 100 I=1,K
    SINTH=SIN(THH)
    COSTH=COS(THH)
    X0=XCENT+RZERO*COSTH
    Y0=YCENT+RZERO*SINTH
    X1=XCENT+RMAX*COSTH
    Y1=YCENT+RMAX*SINTH
    CALL DASHLIN(X0,Y0,X1,Y1,,1)
    THH=THETA1
100 CONTINUE
    CALL DASHLIN(XCENT,YCENT,0.,YCENT,.25)
    CALL DASHLIN(XCENT,YCENT,XCENT,YSIZE,.25)
    CALL DASHLIN(XCENT,YCENT,XCENT,0.,.25)
    CALL DASHLIN(XCENT,YCENT,XSIZE,YCENT,.25)
200 CONTINUE
    RETURN
    END



    SUBROUTINE LABEL(ITITLE,LABELX,LABELY,FX,FY)
```

```
      COMMON/TEMP/Z(101)
      COMMON/XYBNDS/XA,XB,YA,YB,XSIZE,YSIZE,HX,HY,
     1XS,XSS,YS,YSS,FXA,FYA
      COMMON/INDICES/M,N,MM,NN,KFUL
      COMMON/CLEVELS/NCL,NLV,CL(50)
      DIMENSION ITITLE(1),LABELX(1),LABELY(1)
      DATA IS/15/
      J = 0
      X = XA
      P = 0
      G = XSIZE-.9
      DO 10 I=1,NN
      XG = XSS*(FX(X)-FXA)
      IF (XG .LT. G) GO TO 4
      X = XB
      XG = XSIZE
      GO TO 5
    4 IF (XG .LT. P) GO TO 10
    5 J = J+1
      Z(J) = XG
      WE DRAW ARROWS TOGETHER WITH NUMBERS AT THE BOTTOM OF THE GRAPH
      CALL SYMBOL(XG,-.07,.12,IS,0.,-1)
      CALL NUMBER(XG-.42,-.40,.14,X,0,2)
      IF (X .EQ. XB) GO TO 11
      P = XG+.9
   10 X = X+XS
      WE ARE LABELLING THE X AXIS
   11 CALL TITLE(0.,-.65,XSIZE,.14,0.,LABELX)
      WE ARE PUTTING A TITLE AT THE TOP OF THE GRAPH
      CALL TITLE(0.,YSIZE+.15,XSIZE,.21,0.,ITITLE)
      WE ARE PUTTING ARROWS AT THE TOP OF THE GRAPH
      DO 12 I=1,J
   12 CALL SYMBOL(Z(I),YSIZE+.07,.12,IS,180.,-1)
      J=0
      Y = YA
      P = 0
      G = YSIZE-.2
      DO 20 I=1,MM
      YG = YSS*(FY(Y)-FYA)
      IF (YG .LT. G) GO TO 14
      Y = YB
      YG = YSIZE
      GO TO 15
   14 IF (YG .LT. P) GO TO 20
   15 J = J+1
      Z(J) = YG
      WE ARE DRAWING ARROWS TOGETHER WITH NUMBERS AT THE LEFT OF GRAPH
      CALL SYMBOL(-.14,YG,.25,IS,270.,-1)
```

```
      CALL NUMBER(-.85,YG+.16,.14,Y,0,2)
      IF (Y .EQ. YB) GO TO 21
      P = YG+.9
   20 Y = Y+YS
C     WE ARE LABELLING THE Y AXIS
   21 CALL TITLE(-.8,0.,YSIZE,.14,90.,LABELY)
      DO 22 I=1,J
C     WE ARE PLACING ARROWS ON THE RIGHT SIDE OF THE GRAPH
   22 CALL SYMBOL(XSIZE+.07,Z(I),.25,IS,90.,-1)
      YI = .5*YSIZE+.1*FLOAT(NCL)
      DY = .4
C     WE ARE LABELLING THE COUNTOUR CURVES ON THE RIGHT OF THE GRAPH
      CALL NUMBER(XSIZE+.2,YI,.28,NCL,0,2HI2)
      CALL SYMBOL(XSIZE+.30,YI,.28,14HCONTOUR LEVELS,0,14)
      YI=YI-DY
      DO 24 I=1,NCL
      CALL SYMBOL(XSIZE+.3,YI+.05,0,20,I-1,0,-1)
      CALL NUMBER(XSIZE+1.0,YI,0.205,CL(I),0,5HE15.5)
   24 YI=YI-DY
      RETURN
      END




      SUBROUTINE INTERP(AM,IDIM)
      DIMENSION AM(IDIM,1)
      COMMON/TEMP/Z(101)
      COMMON/XYBNDS/XA,XB,YA,YB,XG,YG,HX,HY,
     1XS,XSS,YS,YSS,FXA,FYA
      COMMON/INDICES/M,N,MM,NN,KPOL
      ZFUN(V)=A0+A1*V+A2*V**2
      N1 = N-1
      M1 = M-1
      IF (N-NN) 5,16,50
    5 DO 15 I=1,M
      DO 10 J=1,N
   10 Z(J)=AM(I,J)
      K = 1
      XY = XA
      T = HX+XA
      DO 13 J=2,N1
      CALL FIT(J,T,HX,A0,A1,A2)
   12 AM(I,K)=ZFUN(XY)
      K = K+1
      XY = XY+XS
      IF (XY .LE. T) GO TO 12
   13 T=T+HX
```

```
 14 IF (K .GT. NN) GO TO 15
    AM(I,K) = ZFUN(XY)
    K = K+1
    XY = XY+XS
    GO TO 14
 15 CONTINUE
 16 IF (M-MM) 17,30,50
 17 DO 25 I=1,NN
    DO 18 J=1,M
    Z(J) = AM(J,I)
 18 CONTINUE
    K = 1
    XY = YA
    T = HY+YA
    DO 20 J=2,M1
    CALL FIT(J,T,HY,A0,A1,A2)
 19 AM(K,I)=ZFUN(XY)
    K = K+1
    XY = XY+YS
    IF (XY .LE. T) GO TO 19
 20 T=T+HY
 21 IF (K .GT. MM) GO TO 25
    AM (K,I) = ZFUN(XY)
    K = K+1
    XY = XY+YS
    GO TO 21
 25 CONTINUE
 30 RETURN
 50 STOP12
    END
```

```
    SUBROUTINE SCAN(AM,FX,FY)
AM IS THE MATRIX TO BE CONTOURED, MT AND NT ARE ITS X AND Y DIMENSIONS
CL(NLV) IS THE CONTOUR LEVEL,
THE N  (X,Y)  VALUES OF ONE CONTOUR LINE ARE PLOTTED WHEN
 THEY ARE AVAILABLE.
    DIMENSION AM(1)
    COMMON/CLEVELS/NCL,NLV,CL(50)
    COMMON/INDICES/DUM(2),MT,NT,KFOL
    COMMON/CAVIN/DIM,IX,IY,IDX,IDY,ISS,NP,N,CV,IS,ISO,IX0,IY0,BCP,
   1     INX(8),INY(8),REC(800),X(1603),Y(1603)
    TYPE INTEGER REC,DIM
    DATA(INY=0,1,1,1,0,-1,-1,-1)
    DATA(INX=-1,-1,0,1,1,1,0,-1)
    NP = 0
```

```
      ISS = 0
      CV=CL(NLV)
      MT1=MT-1
      NT1 = NT-1
      DO 10 I=1,MT1
      IF (AM(I+1) .LT. CV .OR. AM(I) .GE. CV) GO TO 10
      IX0 = I+1
      IX = I+1
      IY0 = 1
      IY = 1
      IS0 = 1
      IS = 1
      IDX = -1
      IDY = 0
      CALL TRACE(AM,FX,FY)
   10 CONTINUE
      J=MT-DIM
      DO 20 I=1,NT1
      J = J+DIM
      IF (AM(J+DIM) .LT. CV .OR. AM(J) .GE. CV) GO TO 20
      IX0 = MT
      IX = MT
      IY0 = I+1
      IY = I+1
      IDX = 0
      IDY = -1
      IS0 = 7
      IS = 7
      CALL TRACE(AM,FX,FY)
   20 CONTINUE
      J=MT+NT1*DIM+1
      DO 30 I=1,MT1
      J = J-1
      IF (AM(J-1) .LT. CV .OR. AM(J) .GE. CV) GO TO 30
      IX0 = MT-1
      IX = MT-1
      IY0 = NT
      IY = NT
      IDX = 1
      IDY = 0
      IS0 = 5
      IS = 5
      CALL TRACE(AM,FX,FY)
   30 CONTINUE
      J=NT*DIM+1
      DO 40 I=1,NT1
      J = J-DIM
      IF (AM(J-DIM) .LT. CV .OR. AM(J) .GE. CV) GO TO 40
```

```
      IX0 = 1
      IX = 1
      IY0 = NT-1
      IY = NT-1
      IDX = 0
      IDY = 1
      IS0 = 3
      IS = 3
      CALL TRACE(AM,FX,FY)
   40 CONTINUE
      ISS=1
      L = 0
      DO 70 J=2,NT1
      L = L+DIM
      DO 60 I=1,MT1
      L = L+1
      IF (AM(L+1) .LT. CV .OR. AM(L) .GE. CV) GO TO 60
      K = L+1
      DO 50 ID=1,NP
      IF (REC(ID) .EQ. K) GO TO 60
   50 CONTINUE
      IX0 = I+1
      IX = I+1
      IY0 = J
      IY = J
      IDX = -1
      IDY = 0
      IS0 = 1
      IS = 1
      CALL TRACE(AM,FX,FY)
   60 CONTINUE
   70 L=L-MT1
      RETURN
      END




      SUBROUTINE TRACE(AM,FX,FY)
      DIMENSION AM(1)
      COMMON/POLARC/RS,R0,THS,TH0
      COMMON/INDICES/DUM(2),MT,NT,KFOL
      COMMON/XYBNDS/YA,XB,YA,YB,XSIZE,YSIZE,HX,HY,
     1XS,XSS,YS,YSS,FXA,FYA
      COMMON/CAVIN/DIM,IX,IY,IDX,IDY,ISS,NP,N,CV,IS,IS0,IX0,IY0,DCP,
     1     INX(8),INY(8),REC(800),X(1603),Y(1603)
      COMMON/CLEVELS/FCL,NLV,CL(50)
      TYPE INTEGER REC,DIM
```

```
      N=0
      JY = DIM*(IY-1)+IX
      MY = DIM*IDY+IDX+JY
    2 N=N+1
      IF (N .GT. 1600) GO TO 40
      IF (IDX) 5,4,6
    4 X(N) = FLOAT(IY-1)+FLOAT(IDY)*(AM(JY)-CV)/(AM(JY)-AM(DIM*IDY+JY))
      Y(N) = FLOAT(IX-1)
      GO TO 7
    5 NP=NP+1
      REC(NP) = JY
    6 Y(N) = FLOAT(IX-1)+FLOAT(IDX)*(AM(JY)-CV)/(AM(JY)-AM(JY+IDX))
      X(N) = FLOAT(IY-1)
    7 IS=IS+1
    8 IF (IS .LE. 8) GO TO 10
      IS = IS-8
   10 IDX=INX(IS)
      IDY = INY(IS)
      IX2=IX+IDX
      IY2 = IY+IDY
      IR = IDX*IDY
   11 IF(ISS)13,15
   13 IF(IS.NE.ISO.OR.IY.NE.IYO.OR.IX.NE.IXO) GO TO 16
      N = N+1
      X(N) = X
      Y(N) = Y
      GO TO 45
   15 IF (IX2 .EQ. 0 .OR. IY2 .EQ. 0) GO TO 45
      IF ((IX2 .GT. MT) .OR. (IY2 .GT. NT)) GO TO 45
   16 MY=DIM*IDY+IDX+JY
      IF (IR) 19,17,20
   17 IF (CV .GT. AM(MY)) GO TO 2
      IX = IX2
      IY = IY2
   18 IS = IS+5
      JY = MY
      GO TO 8
   19 KY=JY+IDX
      LY = MY-IDX
      GO TO 21
   20 KY=MY-IDY
      LY = JY+IDY
   21 DCP=(AM(JY)+AM(KY)+AM(LY)+AM(MY))*.25
      IF (CV .LE. DCP) GO TO 23
      CALL GETPT(AM(JY))
      GO TO 7
   23 IF (IR .GE. 0) GO TO 25
      IX = IX2
```

```
      IDX = -IDX
      CALL GETPT(AM(KY))
      IY=IY2
      IDY = -IDY
      GO TO 26
   25 IY=IY2
      IDY = -IDY
      CALL GETPT(AM(KY))
      IX=IX2
      IDX = -IDX
   26 IF (CV .LE. AM(MY)) GO TO 18
      CALL GETPT(AM(MY))
      IF (IR .GE. 0) GO TO 30
      IX = IX+IDX
      IDX = -IDX
      GO TO 31
   30 IY=IY+IDY
      IDY = -IDY
   31 IF (CV .GT. AM(LY)) GO TO 34
      IS = IS-1
      JY = LY
      GO TO 10
   34 CALL GETPT(AM(LY))
      IF (IR .GE. 0) GO TO 36
      IY = IY+IDY
      GO TO 7
   36 IX=IX+IDX
      GO TO 7
   40 PRINT 500, CV
   45 IF(KPOL.EQ.0) GO TO 60
      DO 50 I=1,N
      THETA=X(I)*THS+TH0
      R=Y(I)*RS+R0
      X(I)=XSS*(R*COS(THETA)-FXA)
      Y(I)=YSS*(R*SIN(THETA)-FYA)
   50 CONTINUE
      GO TO 80
   60 DO 70 I=1,N
      X(I)=XSS*(FX(X(I)*XS+XA)-FXA)
   70 Y(I)=YSS*(FY(Y(I)*YS+YA)-FYA)
   80 CONTINUE
      CALL SYMBOL(X,Y,.28,NLV-1,0,-1)
      DO 90 I=1,N
      CALL PLOT(X(I),Y(I),2)
   90 CONTINUE
      RETURN
  500 FORMAT(1H0,23HA CONTOUR LINE AT LEVEL,F13.5,
     1 41H WAS TERMINATED BECAUSE IT CONTAINED MORE,
```

```
2 23H THAN 1000 PLOT POINTS.)
  END



  SUBROUTINE GETPT(AM)
  COMMON/CAVIN/DIM, IX,IY,IDX,IDY,ISS,
1 NP,N,CV,IS,IS0,IX0,IY0,DCP,
2     INX(8),INY(8),REC(800),X(1603),Y(1603)
  N=N+1
  B = AM-DCP
  IF (B) 2,1
1 V=.5
  GO TO 3
2 V = .5*(AM-CV)/B
3 X(N) = FLOAT(IY-1)+FLOAT(IDY)*V
  Y(N) = FLOAT(IX-1)+FLOAT(IDX)*V
  RETURN
  END



  SUBROUTINE FIT(I,X,H,C,B,A)
  COMMON/TEMP/Z(101)
  W=0.5*(Z(I+1)-Z(I-1))/H
  A=0.5*(Z(I+1)+Z(I-1)-Z(I)-Z(I))/H**2
  B = W-2.*X*A
  C=Z(I)+X*(X*A-W)
  RETURN
  END



  SUBROUTINE ARC(X0,Y0,R,TH0,TH1,IDASH)

DRAWS AN ARC OF RADIUS R ABOUT (X0,Y0) FROM THETA=TH0 TO
THETA=TH1, TH0.LT.TH1.  IF IDASH.EQ.0, THE ARC WILL BE SOLID.
IF IDASH.NE.0, THE ARC WILL BE DASHED.
X0, Y0, AND R ARE IN FLOATING POINT INCHES.
TH0 AND TH1 ARE IN RADIANS.

  DELTH=2.*ASIN(.05/R)
  THETA =TH0
  X=X0+R*COS(THETA)
  Y=Y0+R*SIN(THETA)
```

```
      X1=X0+R*COS(TH1)
      Y1=Y0+R*SIN(TH1)
      IPEN=3
    1 CALL PLOT(X,Y,IPEN)
      THETA=THETA+DELTH
      IPEN=5-IPEN
      IF (IDASH .NE. 0) GO TO 2
      IPEN=2
    2 X=X0+R*COS(THETA)
      Y=Y0+R*SIN(THETA)
      IF (THETA .LT. TH1) GO TO 1
      CALL PLOT(X1,Y1,IPEN)
      RETURN
      END



      SUBROUTINE TITLE(X,Y,SIZE,HEIGHT,ANGLE,ITEXT)
      DIMENSION ITEXT(1)
      DATA SIX7,TWO7/.857142857,.285714286/
      CHSIZE=HEIGHT*SIX7
      MAXCHS=SIZE/CHSIZE
      NUMCHS=NCHARS(ITEXT,MAXCHS)
      START=.5*(SIZE-CHSIZE*FLOAT(NUMCHS)+TWO7*HEIGHT)
      TH=ANGLE*.0174533
      CALL SYMBOL(X+START*COS(TH),Y+START*SIN(TH),HEIGHT,ITEXT,ANGLE,
     1   NUMCHS)
      RETURN
      END



      FUNCTION NCHARS(ITEXT,MAXCHS)
      DIMENSION ITEXT(1)
      MAXWDS=MAXCHS/10+1
      DO 1 I=1,MAXWDS
      K=ITEXT(I).AND.7777B
      IF(K.EQ.0)GO TO 2
    1 CONTINUE
      I=MAXWDS
    2 NUMCHS=10*I
      DO 3 J=1,I
      L=I-J+1
      KTEST=ITEXT(L)
      DO 3 M=1,10
      K=KTEST.AND.77B
```

```
      IF((K.NE.0).AND.(K.NE.55B))GO TO 4
      KTEST=ISHIFT(KTEST,54)
      NUMCHS=NUMCHS-1
    3 CONTINUE
    4 NUMCHS=MIN0(NUMCHS,MAXCHS)
      NCHARS=NUMCHS
      RETURN
      END



      SUBROUTINE DASHLIN(X0,Y0,X1,Y1,DASH)
C
C  DRAWS A DASHED LINE FROM (X0,Y0) TO(X1,Y1).  THE DASHES AND SPACES
C  BETWEEN THEM ARE APPROXIMATELY OF LENGTH -DASH-.  THIS LENGTH IS
C  ADJUSTED SUCH THAT THE LINE IS COMPOSED OF EQUAL LENGTH DASHES,
C  AND BEGINS AND ENDS WITH A DASH.
C  ALL PARAMETERS ARE IN FLOATING POINT INCHES.
C
      XNOW=X0
      YNOW=Y0
      DX=X1-X0
      DY=Y1-Y0
      D=SQRT(DY*DY+DX*DX)
      NDASH=2*(IFIX(D/DASH)/2)+1
      DX=DX/FLOAT(NDASH)
      DY=DY/FLOAT(NDASH)
      CALL WHERE(XN,YN,STEPS)
      D1=AMAX1(ABS(XN-X0),ABS(YN-Y0))
      D2=AMAX1(ABS(XN-X1),ABS(YN-Y1))
      IF (D2 .GT. D1) GO TO 1
      XNOW=X1
      YNOW=Y1
      DX=-DX
      DY=-DY
    1 IPEN=3
      CALL PLOT(XNOW,YNOW,IPEN)
      DO 2 I=1,NDASH
      XNOW=XNOW+DX
      YNOW=YNOW+DY
      IPEN=5-IPEN
      CALL PLOT(XNOW,YNOW,IPEN)
    2 CONTINUE
      RETURN
      END
```

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author) Courant Institute of Mathematical Sciences New York University	2a REPORT SECURITY CLASSIFICATION not classified 2b GROUP none

3 REPORT TITLE

Frontal Motion in the Atmosphere

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Report September 1970

5 AUTHOR(S) (Last name, first name, initial)

Turkel, Eli L.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
September 1970	140	20

8a CONTRACT OR GRANT NO. N00014-67-A-0467-0016 b. PROJECT NO. NR 062-160 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) IMM 385 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) none

10 AVAILABILITY/LIMITATION NOTICES

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
none	U.S. Navy, Office of Naval Research 207 West 24th St., New York, N.Y.

13 ABSTRACT

The motion of frontal disturbances in the atmosphere is studied based on several nonlinear models proposed by Stoker. In the first model, the air is considered to be an incompressible fluid moving over a plane tangent to the rotating earth. The fluid consists of two layers and the density in each layer is assumed to be constant. The hydrostatic pressure law is then used to reduce this to a two space dimensional model. The boundary between these layers is a contact discontinuity and so instabilities may occur at this frontal surface.

To simplify this model, we assume that the dynamics of the perturbations in the upper warm air layer can be neglected. In this case only the motion of the cold air need be studied. The frontal surface intersects the horizontal ground in a curve, called the front, which is a free boundary for the cold air. Following the procedure of Kasahara, Isaacson and Stoker, we make a numerical study of this model using generalizations of the Lax-Wendroff scheme. The movement of the front is based on following the motion of material particles at the front. This study indicates the development of the occlusion process from an initially sinusoidal frontal pattern. Thus, we show that the qualitative features of the occlusion process

(continued on p. 143)

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, *or (U)*

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

(continued from p. 141)

depend only on the Coriolis and gravitational forces while the thermodynamic processes can be ignored.

Various initial and boundary conditions are considered, and a comparison of their effect on the occlusion process is made. In all cases, the cold front propagates faster than the warm front, and a relatively strong mass convergence flow exists behind the cold front only. A cyclonic circulation pattern also appears near the cold front. These facts suggest the occurrence of severe storms associated with cold fronts, but not with warm fronts in the atmosphere. The methods developed here have application to general free boundary problems in fluid dynamics.

The numerical study of these equations is still quite difficult and so a one space dimensional model is introduced. Numerical comparison with the two dimensional model shows that this simplified theory gives many of the important characteristics of the frontal motion for reasonable lengths of time.

The one dimensional model is then considered for a semi-infinite domain with constant initial and boundary conditions. The solution of this first order hyperbolic system is expanded in a formal perturbation series. The lowest order terms satisfy a quasi-linear homogeneous hyperbolic system of equations. This system can be analyzed by introducing the Riemann invariants as defined by Lax. Necessary and sufficient conditions for the existence of continuous global solutions are given. When these continuous solutions exist, they can be found explicitly for both subsonic and supersonic flows. The higher order systems are all linear nonhomogeneous equations which can be solved explicitly for the terms in the expansion. Comparison of the series, through second order terms, with the numerical solution of the original system shows close agreement away from the boundary. These techniques can be used for all nonhomogeneous quasi-linear systems where the solution to the homogeneous system is known.

Technical Library
Building 131
Aberdeen Proving Ground, Maryland 21005

Defense Documentation Center (20)
Cameron Station
Alexandria, Virginia 22314

Technical Library
Naval Ship Research and
Development Center
Annapolis Division
Annapolis, Maryland 21402

Professor Bruce Johnson
Engineering Department
Naval Academy
Annapolis, Maryland 21402

Library
Naval Academy
Annapolis, Maryland 21402

Professor W. P. Graebel
Department of Engineering
Mechanics
The University of Michigan
College of Engineering
Ann Arbor, Michigan 48104

Professor W. R. Debler
Department of Engineering Mechanics
University of Mechanics
Ann Arbor, Michigan 48108

Dr. Francis Ogilvie
Department of Naval Architecture
and Marine Engineering
University of Michigan
Ann Arbor, Michigan 48108

Professor S. D. Sharma
Department of Naval Architecture
and Marine Engineering
University of Michigan
Ann Arbor, Michigan 48108

Professor W. W. Willmarth
Department of Aerospace Engineering
University of Michigan
Ann Arbor, Michigan 48108

Professor Finn C. Michelsen
Naval Architecture and Marine
Engineering
445 West Engineering Bldg.
University of Michigan
Ann Arbor, Michigan 48104

AFOSR (REM)
1400 Wilson Boulevard
Arlington, Virginia 22204

Dr. J. Menkes
Institute for Defense Analyses
400 Army-Navy Drive
Arlington, Virginia 22204

Professor S. Corrsin
Mechanics Department
The Johns Hopkins University
Baltimore, Maryland 20910

Professor L. S. G. Kovasznay
The Johns Hopkins University
Baltimore, Maryland 20910

Librarian
Department of Naval Architecture
University of California
Berkeley, California 94720

Professor P. Lieber
Department of Mechanical Engineering
University of California
Institute of Engineering Research
Berkeley, California 94720

Professor M. Holt
Division of Aeronautical Sciences
University of California
Berkeley, California 94720

Professor J. V. Wehausen
Department of Naval Architecture
University of California
Berkeley, California 94720

Professor J. R. Paulling
Department of Naval Architecture
University of California
Berkeley, California 94720

Professor E. V. Laitone
Department of Mechanical Engineering
University of California
Berkeley, California 94720

School of Applied Mathematics
Indiana University
Bloomington, Indiana 47401

Commander
Boston Naval Shipyard
Boston, Massachusetts 02129

Director
Office of Naval Research
Branch Office
495 Summer Street
Boston, Massachusetts 02210

Professor M. S. Uberoi
Department of Aeronautical Engineering
University of Colorado
Boulder, Colorado 80303

Naval Applied Science Laboratory
Technical Library
Bldg. 1 Code 222
Flushing and Washington Avenues
Brooklyn, New York 11251

Professor J. J. Foody
Chairman, Engineering Department
State University of New York
Maritime College
Bronx, New York 10465

Dr. Irving C. Statler, Head
Applied Mechanics Department
Cornell Aeronautical Laboratory, Inc.
P. O. Box 235
Buffalo, New York 14221

Dr. Alfred Ritter
Assistant Head, Applied Mechanics Dept.
Cornell Aeronautical Laboratory, Inc.
Buffalo, New York 14221

Professor G. Birkhoff
Department of Mathematics
Harvard University
Cambridge, Massachusetts 02138

Commanding Officer
NROTC Naval Administrative Unit
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor N. Newman
Department of Naval Architecture and
Marine Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor A. H. Shapiro
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor C. C. Lin
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor E. W. Merrill
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor M. A. Abkowitz
Department of Naval Architecture
and Marine Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor G. H. Carrier
Department of Engineering and
Applied Physics
Harvard University
Cambridge, Massachusetts 02139

Professor E. Mollo-Christensen
Room 54-1722
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Professor A. T. Ippen
Department of Civil Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Commander
Charleston Naval Shipyard
U. S. Naval Base
Charleston, South Carolina 29408

A. R. Kuhlthau, Director
Research Laboratories for the
Engineering Sciences
Thorton Hall, University of Virginia
Charlottesville, Virginia 22903

Director
Office of Naval Research
Branch Office
219 Dearborn Street
Chicago, Illinois 60604

Library
Naval Weapons Center
China Lake, California 93557

Library MS 60-3
NASA Lewis Research Center
21000 Brookpark Road
Cleveland, Ohio 44135

Professor J. M. Burgers
Institute of Fluid Dynamics and
Applied Mathematics
University of Maryland
College Park, Maryland 20742

Acquisition Director
NASA Scientific & Technical
Information
P. O. Box 33
College Park, Maryland 20740

Professor Pai
Institute for Fluid Dynamics
and Applied Mathematics
University of Maryland
College Park, Maryland 20740

Technical Library
Naval Weapons Laboratory
Dahlgren, Virginia 22448

Computation & Analyses Laboratory
Naval Weapons Laboratory
Dahlgren, Virginia 22448

Professor C. S. Wells
LTV Research Center
Dallas, Texas 75222

Dr. R. H. Kraichnan
Dublin, New Hampshire 03444

Commanding Officer
Army Research Office
Box CM, Duke Station
Durham, North Carolina 27706

Professor A. Charnes
The Technological Institute
Northwestern University
Evanston, Illinois 60201

Dr. Martim H. Bloom
Polytechnic Institute of Brooklyn
Graduate Center, Dept. of Aerospace
Engineering and Applied Mechanics
Farmingdale, New York 11735

Technical Documents Center
Building 315
U. S. Army Mobility Equipment
Research and Development Center
Fort Belvoir, Virginia 22060

Professor J. E. Cermak
College of Engineering
Colorado State University
Ft. Collins, Colorado 80521

Technical Library
Webb Institute of Naval Architecture
Glen Cove, Long Island, New York 11542

Professor E. V. Lewis
Webb Institute of Naval Architecture
Glen Cove, Long Island, New York 11542

Library MS 185
NASA, Langley Research Center
Langley Station
Hampton, Virginia 23365

Dr. B. N. Pridmore Brown
Northrop Corporation
NORAIR-Div.
Hawthorne, California 90250

Dr. J. P. Breslin
Stevens Institute of Technology
Davidson Laboratory
Hoboken, New Jersey 07030

Mr. D. Savitsky
Stevens Institute of Technology
Davidson Laboratory
Hoboken, New Jersey 07030

Mr. C. H. Henry
Stevens Institute of Technology
Davidson Laboratory
Hoboken, New Jersey 07030

Professor J. F. Kennedy, Director
Iowa Institute of Hydraulic Research
State University of Iowa
Iowa City, Iowa 52240

Professor L. Landweber
Iowa Institute of Hydraulic Research
State University of Iowa
Iowa City, Iowa 52240

Professor E. L. Resler
Graduate School of
Aeronautical Engineering
Cornell University
Ithaca, New York 14851

Professor John Miles
% I.G.P.P.
University of California, San Diego
La Jolla, California 92038

Director
Scripps Institution of Oceanography
University of California
La Jolla, California 92037

Dr. B. Sternlicht
Mechanical Technology Incorporated
968 Albany-Shaker Road
Latham, New York 12110

Mr. P. Eisenberg, President
Hydronautics
Pindell School Road
Howard County
Laurel, Maryland 20810

(2)

Professor A. Ellis
University of California, San Diego
Department of Aerospace & Mech. Engrg. Sci.
La Jolla, California 92037

Mr. Alfonso Alcedan L. Director
Laboratorio Nacional De Hydraulics
Antigui Cameno A. Ancon
Casilla Jostal 682
Lima, Peru

Commander
Long Beach Naval Shipyard
Long Beach, California 90802

Professor John Laufer
Department of Aerospace Engineering
University Park
Los Angeles, California 90007

Professor J. Ripkin
St. Anthony Falls Hydraulic Lab.
University of Minnesota
Minneapolis, Minnesota 55414

Professor J. M. Killen
St. Anthony Falls Hydraulic Lab.
University of Minnesota
Minneapolis, Minnesota 55414

Lorenz G. Straub Library
St. Anthony Falls Hydraulic Lab.
Mississippi River at 3rd Avenue SE.
Minneapolis, Minnesota 55414

Dr. E. Silberman
St. Anthony Falls Hydraulic Lab.
University of Minnesota
Minneapolis, Minnesota 55414

Superintendent
Naval Postgraduate School
Library Code 0212
Monterey, California 93940

Professor A. B. Metzner
University of Delaware
Newark, New Jersey 19711

Technical Library
USN Underwater Weapons &
Research & Engineering Station
Newport, Rhode Island 02840

Technical Library
Underwater Sound Laboratory
Fort Trumbull
New London, Connecticut 06321

Professor J. J. Stoker
Institute of Mathematical Sciences
New York University
251 Mercer Street
New York, New York 10003

Engineering Societies Library
345 East 47th Street
New York, New York 10017

Office of Naval Research
New York Area Office
207 W. 24th Street
New York, New York 10011

Professor H. Elrod
Department of Mechanical Engineering
Columbia University
New York, New York 10027

Society of Naval Architects and
Marine Engineering
74 Trinity Place
New York, New York 10006

Professor S. A. Piascek
Department of Engineering Mechanics
University of Notre Dame
Notre Dame, Indiana 46556

United States Atomic Energy Commission
Division of Technical Information
Extension
P. O. Box 62
Oak Ridge, Tennessee 37830

Miss O. M. Leach, Librarian
National Research Council
Aeronautical Library
Montreal Road
Ottawa 7, Canada

Technical Library
Naval Ship Research and
Development Center
Panama City, Florida 32401

Library
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Avenue
Pasadena, California 91109

Professor M. S. Plesset
Engineering Division
California Institute of Technology
Pasadena, California 91109

Professor H. Liepmann
Department of Aeronautics
California Institute of Technology
Pasadena, California 91109

Technical Library
Naval Undersea Warfare Center
3202 E. Foothill Boulevard
Pasadena, California 91107

Dr. J. W. Hoyt
Naval Undersea Warfare Center
3202 E. Foothill Boulevard
Pasadena, California 91107

Professor T. Y. Wu
Department of Engineering
California Institute of Technology
Pasadena, California 91109

Director
Office of Naval Research
Branch Office
1030 E. Green Street
Pasadena, California 91101

Professor A. Acosta
Department of Mechanical Engineering
California Institute of Technology
Pasadena, California 91109

Naval Ship Engineering Center
Philadelphia Division
Technical Library
Philadelphia, Pennsylvania 19112

Technical Library (Code 249B)
Philadelphia Naval Shipyard
Philadelphia, Pennsylvania 19112

Professor R. C. Mac Camy
Department of Mathematics
Carnegie Institute of Technology
Pittsburgh, Pennsylvania 15213

Dr. Paul Kaplan
Oceanics, Inc.
Plainview, Long Island, New York 11803

Technical Library
Naval Missile Center
Point Mugu, California 93441

Technical Library
Naval Civil Engineering Lab.
Port Hueneme, California 93041

Commander
Portsmouth Naval Shipyard
Portsmouth, New Hampshire 03801

Commander
Norfolk Naval Shipyard
Portsmouth, Virginia 23709

Professor F. E. Bisshopp
Division of Engineering
Brown University
Providence, Rhode Island 02912

Dr. L. L. Higgins
TRW Space Technology Labs, Inc.
One Space Park
Redondo Beach, California 90278

Redstone Scientific Information Center
Attn: Chief, Document Section
Army Missile Command
Redstone Arsenal, Alabama 35809

Dr. H. N. Abramson
Southwest Research Institute
8500 Culebra Road
San Antonio, Texas 78228

Editor
Applied Mechanics Review
Southwest Research Institute
8500 Culebra Road
San Antonio, Texas 78206

Librarian
Naval Command Control Communications
Laboratory Center
San Diego, California 92152

Library & Information Services
General Dynamics-Convair
P. O. Box 1128
San Diego, California 92112

Commander (Code 246P)
Pearl Harbor Naval Shipyard
Box 400
FPO San Francisco, California 96610

Technical Library (Code H245C-3)
Hunters Point Division
San Francisco Bay Naval Shipyard
San Francisco, California 94135

Dr. A. May
Naval Ordnance Laboratory
White Oak
Silver Spring, Maryland 20910

Fenton Kennedy Document Library
The Johns Hopkins University
Applied Physics Laboratory
8621 Georgia Avenue
Silver Spring, Maryland 20910

Librarian
Naval Ordnance Laboratory
White Oak
Silver Spring, Maryland 20910

Dr. Bryne Perry
Department of Civil Engineering
Stanford University
Stanford, California 94305

Professor Milton Van Dyke
Department of Aeronautical Engineering
Stanford University
Stanford , California 94305

Professor E. Y. Hsu
Department of Civil Engineering
Stanford University
Stanford, California 94305

Dr. R. L. Street
Department of Civil Engineering
Stanford University
Stanford, California 94305

AIR 5301
Naval Air Systems Command
Department of the Navy
Washington, D.C. 20360

AIR 604
Naval Air Systems Command
Department of the Navy
Washington, D.C. 20360

Dr. John Craven (PM1100)
Deep Submergence Systems
Project
Department of the Navy
Washington, D.C. 20360

Code 522
Naval Ship Research and
Development Center
Washington, D.C. 20007

Commander
Naval Oceanographic Office
Washington, D.C. 20390

Chief of Research and Development
Office of Chief of Staff
Department of the Army
The Pentagon
Washington, D.C. 20310

Code 6342A
Naval Ship Systems Command
Department of the Navy
Washington, D.C.

Code 468
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

Director
U.S. Naval Research Laboratory
Code 6170
Washington, D.C. 20390

Code 473
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

Code 6100
Naval Ship Engineering Center
Department of the Navy
Washington, D.C. 20360

Mr. Ralph Lacey (Code 6114)
Naval Ship Engineering Center
Department of the Navy
Washington, D.C. 20360

Dr. A. S. Iberall, President
General Technical Services, Inc.
451 Penn Street
Yeadon, Pennsylvania 19050

Dr. H. Cohen
IBM Research Center
P. O. Box 218
Yorktown Heights, New York 10598

Director 25
Naval Research Lab.
Code 2029 (ONR2)
Washington, D.C. 20390
Attn: Library

Stanford University
Department of Mathematics
Stanford, California

Office of Naval Research 2
Department of the Navy (Code 432)
Washington, D.C. 20360

Office of Naval Research
San Francisco Area
50 Fell Street
San Francisco, California 94102

Library 2
Courant Institute of Mathematical
Sciences
New York University
New York, New York 10012

Science Library
New York University
Brown 120
New York, New York 10012

Prof. O. M. Phillips
The Johns Hopkins University
Baltimore, Md. 20910

Professor S. Eskinazi
Department of Mechanical Engineering
Syracuse University
Syracuse, New York 13210

Professor R. Pfeffer
Florida State University
Geophysical Fluid Dynamics Institute
Tallahassee, Florida 32306

Professor J. Foa
Department of Aeronautical Engineering
Rennsselaer Polytechnic Institute
Troy, New York 12180

Professor R. C. Di Prima
Department of Mathematics
Rennsselaer Polytechnic Institute
Troy, New York 12180

Dr. M. Sevik
Ordnance Research Laboratory
Pennsylvania State University
University Park, Pennsylvania 16801

Professor J. Lumley
Ordnance Research Laboratory
Pennsylvania State University
University Park, Pennsylvania 16801

Dr. J. M. Robertson
Department of Theoretical and
Applied Mechanics
University of Illinois
Urbana, Illinois 61803

Shipyard Technical Library
Code 130L7 Building 746
San Francisco Bay Naval Shipyard
Vallejo, California 94592

Code L42
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 800
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 2027
U. S. Naval Research Laboratory
Washington, D.C. 20390 (6)

Code 438
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360 (3)

Code 513
Naval Ship Research and
Development Center
Washington, D.C. 20007

Science & Technology Division
Library of Congress
Washington, D.C. 20540

ORD 913 (Library)
Naval Ordnance Systems Command
Washington, D.C. 20360

Code 6420
Naval Ship Engineering Center
Concept Design Division
Washington, D.C. 20360

Code 500
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 901
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 520
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 0341
Naval Ship Systems Command
Department of the Navy
Washington, D.C. 20360

Code 2052 (Technical Library)
Naval Ship Systems Command
Department of the Navy
Washington, D.C. 20360

Mr. J. L. Schuler (Code 03412)
Naval Ship Systems Command
Department of the Navy
Washington, D.C. 20360

Dr. J. H. Huth (Code 031)
Naval Ship Systems Command
Department of the Navy
Washington, D.C. 20360

Code 461
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

Code 530
Naval Ship Research and
Development Center
Washington, D.C. 20360

Code 466
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

Office of Research and Development
Maritime Administration
441 G. Street, NW.
Washington, D.C. 20235

Code 463
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

National Science Foundation
Engineering Division
1800 G. Street, NW.
Washington, D.C. 20550

Dr. G. Kulin
National Bureau of Standards
Washington, D.C. 20234

Department of the Army
Coastal Engineering Research Center
5201 Little Falls Road, NW.
Washington, D.C. 20011

Code 521
Naval Ship Research and
Development Center
Washington, D.C. 20007

Code 481
Chief of Naval Research
Department of the Navy
Washington, D.C. 20390

Code 421
Chief of Naval Research
Department of the Navy
Washington, D.C. 20360

Commander
Naval Ordnance Systems Command
Code ORD 035
Washington, D.C. 20360

Librarian Station 5-2
Coast Guard Headquarters
1300 E. Street, NW.
Washington, D.C. 20226

Division of Ship Design
Maritime Administration
441 G. Street, NW.
Washington, D.C. 20235

HQ USAF (AFRSTD)
Room 1D 377
The Pentagon
Washington, D.C. 20330

Commander
Naval Ship Systems Command
Code 6440
Washington, D.C. 20360

Code 525
Naval Ship Research and
Development Center
Washington, D.C. 20007

Dr. A. Powell (Code 01)
Naval Ship Research and
Development Center
Washington, D.C. 20007

Director of Research Code RR
National Aeronautics & Space Admin.
600 Independence Avenue, SW.
Washington, D.C. 20546

Commander
Naval Ordnance Systems Command
Code 03
Washington, D.C. 20360

Code ORD 05411
Naval Ordnance Systems Command
Washington, D.C. 20360

DATE DUE	BORROWER'S NAME
AUG 26 [illegible]	D. Hootsma
APR 30 [illegible]	Angelia
1.19.88	ILL - CREIGHTON U

DATE DUE

AUG 28 1973			
APR 30 1981			
~~ILL - 1.19.88~~			